

Vortex 10

A stochastic simulation of the extinction process

Version 10.5.0.0



[Begin a New Project](#)

Open a Project: [Existing](#) | [Recent](#)

[Quit](#)

User's Manual

Written by Robert C. Lacy,
Philip S. Miller, and Kathy Traylor-Holzer
30 March 2021 update

Copyright © 2021 IUCN SSC Conservation Planning Specialist Group & Chicago Zoological Society

Citations

The *VORTEX* program should be cited as:

Lacy, R.C., and J.P. Pollak. 2021. Vortex: A Stochastic Simulation of the Extinction Process. Version 10.5.5. Chicago Zoological Society, Brookfield, Illinois, USA.

This manual should be cited as:

Lacy, R.C., P.S. Miller, and K. Traylor-Holzer. 2021. Vortex 10 User's Manual. 30 March 2021 update. IUCN SSC Conservation Planning Specialist Group, and Chicago Zoological Society, Apple Valley, Minnesota, USA.

Citations for the concepts, algorithms, and program flow – although a little out of date because they describe version 9 or earlier (but still mostly are accurate descriptions of the standard *VORTEX* model) – include:

- Lacy, R.C. 1993. *VORTEX*: A computer simulation model for Population Viability Analysis. *Wildlife Research* 20:45-65.
- Lacy, R.C. 1993/1994. What is Population (and Habitat) Viability Analysis? *Primate Conservation* 14/15:27-33.
- Lacy, R.C. 2000. Considering threats to the viability of small populations. *Ecological Bulletins* 48:39-51.
- Lacy, R.C. 2000. Structure of the *VORTEX* simulation model for population viability analysis. *Ecological Bulletins* 48:191-203.

Much of the material from the first two papers is incorporated into an Appendix.

Please [Register](#) as a Vortex user (can be completely anonymous) to provide us with information about how the SCTI software is being used and how we can better serve species conservation.

Using this manual

This document provides an initial description of how to use *VORTEX* version 10. The main part of the manual focuses on the details of using the *VORTEX* user interface. Text that you would type as input is shown in **Bold Microsoft San Serif** font. There are also appendices that provide more background about the structure of the program and its use for Population Viability Analysis. The references are still being updated and expanded, and more will be added. Boxes throughout the manual provide helpful hints, examples or case studies, and other side comments. Table of Contents entries are hyperlinked to those pages.

Table of Contents

| | |
|---|----|
| Table of Contents | 3 |
| Getting Started..... | 7 |
| Installation..... | 7 |
| <i>EDDY</i> : a lite version of Vortex for educational use..... | 7 |
| A Note about Cost..... | 8 |
| Starting <i>VORTEX</i> | 9 |
| Running <i>VORTEX</i> from a Command prompt..... | 11 |
| Input files..... | 12 |
| Output files | 12 |
| Navigating around <i>VORTEX</i> | 12 |
| A caution about models with long run-time | 13 |
| Getting updates..... | 13 |
| Creating your Vortex Project – Entering data | 14 |
| Project Settings | 14 |
| Special Options..... | 15 |
| Simulation Input | 20 |
| Scenario Settings | 22 |
| Species Description | 28 |

| | |
|--------------------------------|----|
| State Variables..... | 35 |
| Dispersal..... | 40 |
| Reproductive System..... | 44 |
| Reproductive Rates..... | 49 |
| Mortality Rates | 53 |
| Catastrophes | 55 |
| Mate Monopolization | 57 |
| Initial Population Size | 58 |
| Carrying Capacity | 60 |
| Harvest | 63 |
| Supplementation..... | 65 |
| Genetics..... | 67 |
| Running your simulation | 78 |
| Text Output..... | 82 |
| Input Summary | 82 |
| Deterministic Results | 83 |
| Output Summary | 85 |
| Output Tables | 92 |

| | |
|---|-----------|
| ST Tables..... | 93 |
| Other Output Files | 94 |
| Tables and Graphs | 95 |
| Standard Graphs..... | 96 |
| Custom Plots | 101 |
| Project Report..... | 104 |
| Sensitivity Tests (ST) | 105 |
| Graphing ST results..... | 113 |
| Using Functions in <i>VORTEX</i> | 118 |
| Primary changes in Functions from <i>VORTEX</i> 9 to <i>VORTEX</i> 10 | 119 |
| Specification of Demographic Rates as Functions..... | 121 |
| Using the Function Editor..... | 122 |
| Variables and Operators Available for Use in Functions | 124 |
| Trigonometric Operators..... | 128 |
| Operations on Lists | 130 |
| Accessing histories of State Variables | 133 |
| Using Random Numbers in Functions | 133 |
| Notes Regarding Function Syntax and Use..... | 134 |
| Using Functions to Examine Genetic Evolution | 136 |

| | |
|--|-----|
| Using VMacros in Vortex | 137 |
| Using R scripts in Vortex | 140 |
| Examples of Rate Functions..... | 142 |
| | |
| An Overview of Population Viability Analysis Using <i>VORTEX</i> | 151 |
| Primary changes from Vortex 9 to Vortex 10 | 168 |
| Literature Cited..... | 174 |

Getting Started

Installation

Please [Register](#) as a Vortex user (can be completely anonymous) to provide us with information about how the SCTI software is being used and how we can better serve species conservation.

Download the Vortex10Installation.msi from scti.tools/vortex, and run the msi to install the program. The program and your projects folder can be installed anywhere, although your projects will need to be installed in a folder where you have full read/write access. (On many systems, especially if managed by an IT department, users are not authorized to access files in folders such as “Program Files”, or sometimes in any folder except those within “MyDocuments”.)

On some old Windows systems, the Vortex10Installation.msi file cannot be started by just double-clicking on that installation file. In that case, the user can install *VORTEX* by running the Setup.exe that is included with the .msi file in a zip file available on the website. (The two files should first be extracted from the zip file.)

A QuietInstallation is available (<https://scti.tools/software/Vortex10QuietInstallation.msi>) for users (primarily system administrators) who wish to install *VORTEX* into default directories without any need for user interaction during the installation. This can be useful if you are installing on 100 computers for a classroom.

EDDY: a lite version of Vortex for educational use

Over time, *VORTEX* has been enhanced to provide modeling of increasingly complex population dynamics. While the newer features are useful for many situations, beginning users and casual users will generally not have the need for the most complex modeling and should probably avoid trying to use some features until they become familiar with the more basic population model in *VORTEX*. A version of *VORTEX* – named *EDDY* – that hides features that provide the more flexible but potentially confusing components of *VORTEX* is included with the installation.

The user interface for *EDDY* does not include the pages for State Variables or Genetics, omits some of the options on other input pages, does not include tools for editing Functions, does not list Special Options, and does not provide automated Sensitivity Testing. When *EDDY* is run, the full capabilities of *VORTEX* are available, but hidden from the user. The program will use the default values for all input variables not shown to the user. (For example, the genetics model will simulate one neutral locus, and inbreeding depression is assumed to be due half to recessive lethals and half to heterosis.) However, if a project that was created in the full *VORTEX* is opened in *EDDY*, and if any of the more complex settings that are hidden in *EDDY* had been specified in the *VORTEX* project input, those options will be invoked when the project is run within *EDDY*, and they will be retained and saved when the project is saved. This can be useful for teaching; for

example, special Genetics or other options might be specified in a sample project file (created in *VORTEX*) that is then used as the basis for a class exercise using the *EDDY* interface.

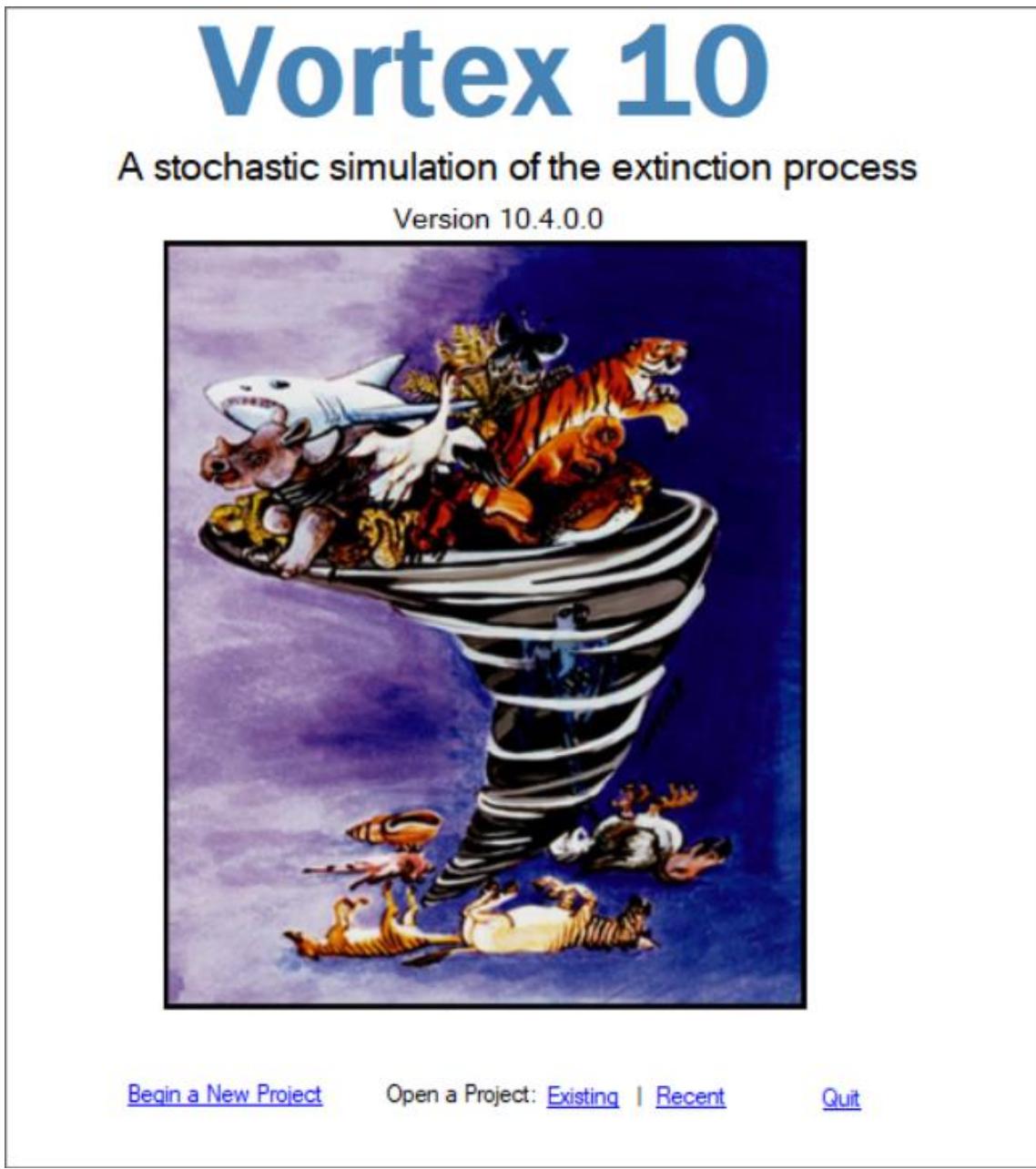
A Note about Cost

VORTEX is provided free of charge because of the commitment of the Chicago Zoological Society to making it widely available to further biodiversity conservation. Similarly, the manual, developed in partnership with the CPSG, is provided for downloading because the CPSG cares about saving species and their habitats. However, the initial development and continuing improvement of the software and manual do represent a significant commitment by these conservation organizations. The rate at which improvements can be made is determined by the resources available to support that work.

If your budget allows it, please consider making a donation to support the further development of *VORTEX*. If you find the software to be especially valuable to you, consider donating perhaps US\$100 (a wild guess about the investment of resources per user that have gone into *VORTEX*), or more or less as you feel is appropriate, to the Chicago Zoological Society. If you find the manual to be especially helpful, consider donating to the CPSG. As a side benefit to US tax-payers, donations to either the Chicago Zoological Society or the CPSG are tax-deductible. Donations to the Chicago Zoological Society should be as a check written to the Chicago Zoological Society, sent to "SCTI donation, CET, Chicago Zoological Society, Brookfield, IL 60513 USA". Donations to the CPSG should be sent to "SCTI donation, CPSG, 12101 Johnny Cake Ridge Road, Apple Valley, MN 55124, USA".

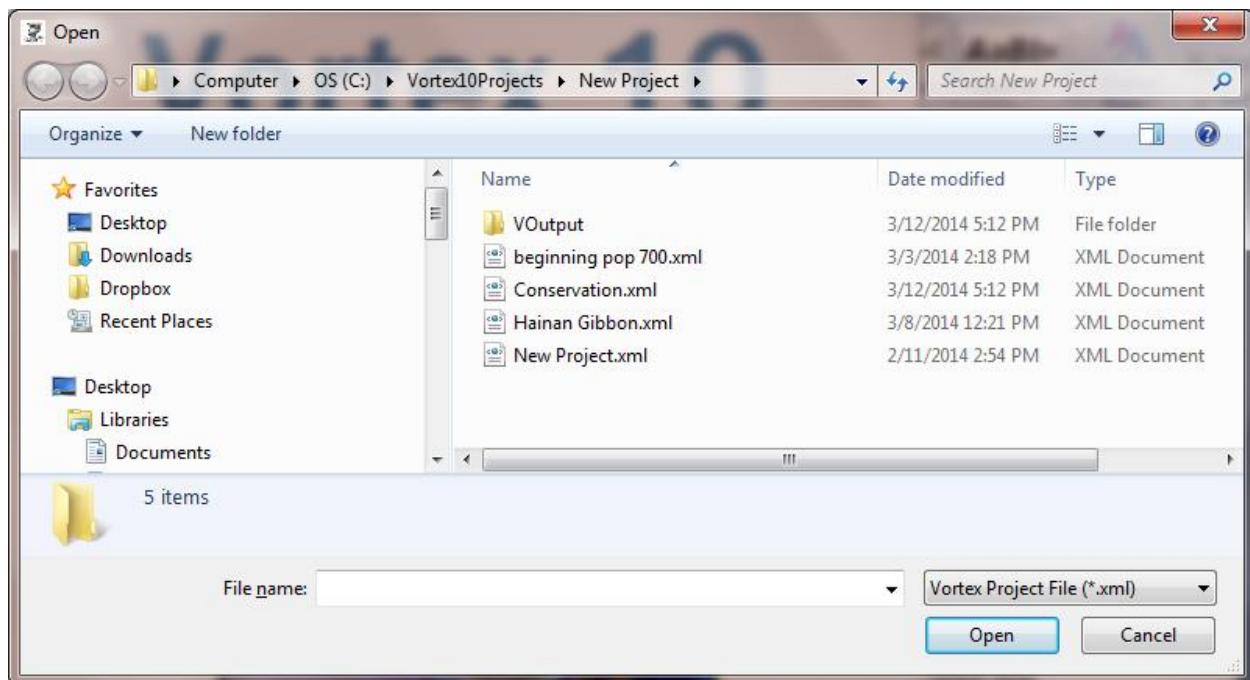
Starting *VORTEX*

Most users will create and run *VORTEX* population simulations from within the *VORTEX* user interface, although the functionality of the *VORTEX* simulation is available within program libraries (dll files) that can be accessed by other interfaces (such as *METAMODEL MANAGER*, available at <https://scti.tools/metamodelmanager/>) and *VORTEX* Project files created in any software can be run from a command line version of *VORTEX* (see below). The opening window provides options to start a new Project (which will initially have a default scenario), open an existing previously saved Project, or open a list of recently opened Projects for quicker access.



If you start a new Project with the default scenario, it is important to go through every input page to change the initial default input values to something that is appropriate for your case. The default values let you run a test scenario just to see that *VORTEX* is working for you, and the defaults also are often useful indicators of the magnitude and format of a typical input value (e.g., a percent, or an integer, or a number from 0 to 1). However, if you forget to change the default values on an input page while you are creating a scenario for your analysis, you may not be getting the scenario that you had intended.

When you select an existing Project, the Open dialog will initially go to the last folder in which you had worked, but you can navigate to other folders. The default project file type searched for will be an xml file, but the dropdown list also allows you to instead search for *VORTEX* 9 vpj files.



Vortex cannot have several projects open at once. However, you can start several instances of Vortex and have each one running a different project. On modern multi-core processors, several concurrent Vortex's may run almost as quickly as a single Vortex runs its simulation. However, don't open the same project simultaneously in several instances of Vortex, because they will likely run into conflicts while writing to output files. Also, it is best not to run multiple projects in the same folder at the same time, because a conflict can sometimes occur during the writing of temporary files. It is safest to run the multiple projects in different folders.

Vortex does have a utility for running multiple scenarios of a project simultaneously, by partitioning the work among different CPUs on the machine. (See the section on [Running Your Simulation](#) for more information about Multi-Vortex runs.)

Running *VORTEX* from a Command prompt

It is sometimes useful to be able to run *VORTEX* projects from a command prompt, rather than from within the user interface. The most common need for this is when someone writes an external shell program (for example, in R or another language) to generate a large number of scenarios for testing – for example, to run sensitivity tests that use a different sampling scheme than the ones available within the ST module of the interface.

To run *VORTEX* from a command line, instead of starting Vortex10.exe, use the program Vortex10Command.exe that is included in the installation. Then, to run a *VORTEX* model from an OS command, use syntax as in (all on one line):
"C:\Program Files (x86)\Species Conservation Toolkit Initiative\Vortex10\Vortex10Command.exe"
"C:\Vortex10Projects\ZPG.xml" "1,2"

Thus, the name of the exe is specified, followed by a parameter that is your project file, and then a list of which scenarios are to be run. You need to specify whatever paths contain your program and project files (and it is safest to fully specify paths rather than assuming that a default location will work). The additional (and optional) parameter **"1,2"** in the example above specifies that the 1st and 2nd scenarios in the project file are to be run. A space delimiter can also be used instead of the comma between scenario numbers. Enclose the string of scenario numbers within quotes, so that it is read as a single command line parameter rather than as a series of parameters. The quotes around the program and project file names avoid problems with spaces, punctuation, and '\' (which has special meaning in computer strings) in the paths. (But if the above syntax does not work for you, try it with '/' symbols instead.) If you do not provide on the command line a list of the scenarios to be run, then all scenarios in the project will run, except for suites of ST samples (which are not stored within the project file as fully specified scenarios).

An important note about regional data formats: Vortex will use whichever numeric data format (e.g., **0.50** or **0,50** for $\frac{1}{2}$) is appropriate for the regional settings on the computer. However, Project files cannot be transferred between computers that use different numeric data formats, unless they are first manually edited to convert the data.

There is also a command line version of *VORTEX* that can make use of multiple CPUs to run scenarios simultaneously, for greater speed when doing large sets of analyses. (See the section on [Running Your Simulation](#) for more information about Multi-Vortex runs.) To run multiple scenarios simultaneously from a command prompt, use syntax as in:

"C:\Program Files (x86)\Species Conservation Toolkit Initiative\Vortex10\MultiVortexCmd.exe"
"C:\Vortex10Projects\ZPG.xml" "1 2 3 4 5 6 7 8 M4"

The extra code of **M4** in the list of scenarios tells *VORTEX* to run 4 scenarios at a time, each using its own CPU. (If there are fewer than that many CPUs on the system, then Windows will have some scenarios share CPUs, which will likely be a less efficient use of computer resources.) The **M4** code can be at any location in the scenario list. If there is no **M** code specified, then *VORTEX* will try to determine the optimal number of CPUs to use, based on the processor on your computer. An **N** code can also be inserted instead of the list of scenarios. **N6** would indicate that the first 6 scenarios of the project are to be run. This can be useful if the project has 10s or even

100s of scenarios and you do not want to list them all on the command line. Thus, the above example would be the same as specifying a second parameter string of “**M4 N8**”.

Input files

The *VORTEX* project files will be saved in .xml format. Although it is easiest to enter input values from the user interface, the project files that specify input values can be edited directly in a browser or text editor, *if great care is taken to format everything in the file properly*. This can be useful if *VORTEX* scenarios are being created by an external shell program.

VORTEX 10 can still read old *VORTEX* 9 vpj project files. However, because the syntax of some variables used in functions has been changed (see below), some editing of input values may be necessary to get a *VORTEX* 9 project to run correctly in *VORTEX* 10. Files saved in *VORTEX* 10 format cannot be read subsequently by *VORTEX* 9. Sensitivity Testing specifications in a *VORTEX* 9 vpj file will not be read into *VORTEX* 10, because the ST module has been completely changed. *After reading any Vortex9 vpj files into Vortex10, go through all input sections to be certain that values were translated correctly between versions, and fix any that were not.*

Conversion hint: The change to .xml files was made because they can be more forgiving of possible errors in the file format, and in the future this will make it easier to open old projects in upgraded versions of Vortex. Be careful, however, that after you save a project in the .xml format you will later want to open that new .xml file and not the prior .vpj file from Vortex9.

A caution about directly editing the Project file: XML uses the symbols > and < in a special way to demarcate data fields. Therefore, you can't type a function that uses these symbols into the XML file, or you will corrupt the file. Within XML files, the greater than and less than symbols are converted to the codes “>” and “<”. *VORTEX* does this conversion automatically when it saves and then reads the .xml Project file but you will need to do it if you edit the .xml file outside of the *VORTEX* interface.

Output files

The output data that are displayed in text, tables, and graphs are all saved in files that are placed in the VOutput subfolder of your project folder. Mostly the files are formatted so that they can be imported into Excel and the data will all be in columns with appropriate headers. This allows the data to be analyzed easily by statistical or other software outside of *VORTEX*.

Navigating around *VORTEX*

VORTEX has four basic levels of navigation. At the top of the *VORTEX* window are a few high-level menus for File operations, running the Simulation, and asking for Help. The Help command, which can be invoked from the Help menu, from the ‘?’ icon, or from any window by hitting **F1**, will open the manual in Acrobat Reader (which must exist on the computer for the

manual to be used). The Help command will try to go to the page of the manual that is appropriate for where you were in the program, but you can browse through the file or search (with **ctrl-F**) for key words to find the help that you need. The SCTI.tools website includes a FAQ section to which users can post questions and answer questions posted by others.

Under the top menus is a row of buttons that provide quick ways to get to many of the same features that are available in the menus. Below these are a series of five tabbed sections. Each of these is explained (to some extent) below. Within these tabbed sections, there are various methods to navigate around parts of the section.

On most screens, if you hover the mouse over a label (e.g., for an icon or menu item, or for an input variable or input table), *VORTEX* will often pop up short text (a “tooltip”) that explains the button or variable further. When you hover over a data entry box, it will often pop up a tooltip that states what kind of data is expected. One clue as to if a data entry box expects a text or can accept a function (in place of a number) is that often the box will be relatively small if you must enter a simple number, but it will be long if you should enter text or can enter a function.

To enter a function in place of a constant number in *VORTEX*, preface your function with the **=** sign, as in **=40+10*(A>3)** to set a rate to be 40 for individuals up through 3 years of age, and a rate of 50 thereafter.

A caution about models with long run-time

If an analysis requires a very long time to complete, it is possible that Windows will enter a sleep (suspend) mode when it does not detect any activity from the user. (MS Windows is sometimes too stupid to detect that a program is busy working.) You can prevent this by setting Windows power-management settings to not enter sleep mode when it does not detect user activity. Screen-saver mode is OK, as it will not stop *VORTEX* from working, and screen saver mode (or otherwise disabling the screen display) may actually allow *VORTEX* to run a little faster. Just do not let Windows enter “Sleep” or “Suspend” mode when *VORTEX* is running; otherwise, *VORTEX* will indeed suspend its work.

Getting updates

In the Help menu is an option to Check for Updates. This will go to the *VORTEX* website and compare the most recently released version against the version that you are using. If a more recent version is available, then a window will open that provides the links to get the latest installation. It is recommended that you check for updates occasionally, especially if you encounter a problem that you think may be a bug. (That bug might already have been fixed!)

The installation of *VORTEX* will place into the *VORTEX* program folder a document file (Vortex10ChangeLog.doc) that lists the primary changes that have been made recently to the *VORTEX* program. This change log is also available on the website.

Creating your Vortex Project – Entering data

Project Settings

Project Name

A Project Name can be almost any label that you want. However, it will be used in the filenames for output files, so do not use any characters (such as '\', '/', '*', or '?') that might be invalid within a filename. You might want to use relatively short Project Names to avoid very long file names. Windows places a limit of 256 characters on any path (folder + filename), and *VORTEX* will try to warn you if the filenames it needs to create for output (consisting of the path + project name + scenario name + population name + extension) will exceed what Windows allows.

Project Notes

In the Project Settings, you have the option to add any notes that you wish to make to document the Project. It is also often helpful to include within these Project Notes the names of the user team that is developing the project (this documentation may be especially helpful in workshop or classroom settings).



We strongly encourage you to take the time to add notes to your Project at this window, during specification of input parameters, and in your Project Report. The extra few minutes you spend documenting your work may save you and others many hours of work later, when you try to remember what information and logic was used to create the project. Unfortunately, many PVAs are not reproducible because the authors did not fully document their work.

The screenshot shows the 'Project Settings' tab selected in the Vortex 10 software interface. The 'Project Name' field contains 'New Project'. Below it is a 'Project Notes and Users' section with a text area containing 'My notes about the project ...'. To the right are several configuration sections: 'Special Options' (checkboxes for graph and message visibility, breeding prioritization, MK calculation, and population variable updates); 'Translocation options' (checkboxes for holding population and random selection); and a checkbox for 'Use undocumented options' with a dropdown menu showing 'MKD GDM2'.

Special Options

Special Options, available on the Project Settings page:

- Do not show graphs during iterations. It is fun to watch the simulations proceed, but if you won't be watching, then the simulations will run faster if you turn off the graphs of changing population sizes.
- Do not show any messages to the user while running. This option can be useful if many scenarios are running when the computer will not be attended.
- Use an Individual State variable to specify the order in which individuals are chosen for pairing. This option works similarly to the Mean Kinship option in Genetics, but it allows the user to create an ISvar that determines the priority for breeding. For example, pairs can be made first for older animals, or more dominant ones, or the ones with the lowest inbreeding. This option would normally be used with the Breed to K option in Genetic Management, as otherwise all females would be available to breed (albeit in the specified order). The priority for pairing will be from the lowest value of the specified ISvar (first) to the highest (last). If a negative is placed before the ISvar code in the box, then the breeding priority will be dynamic – meaning that the ISvar will be updated with its Transition function after each pairing is made. In this way, the breeding priority can be set to a function of, for example, the number of pairs already produced (PAIRS variable).
- Update population variables before each individual's ISupdate. As of version 10.5.2, the sequence of updating ISvars via the ISvar Transition function is that Vortex steps through each population, and then steps through each individual in the populations. The population-level variables, such as N, M, F, IMEAN1, IMAX1, and ITOT1 are specified before the ISupdates are begun for each population. Thus, they are not updated dynamically while the ISupdate is stepping through the population. This Special Option allows you to specify instead that the population-level variables are recalculated before each individual's ISvars are updated. This can be important if the ISvars are changing the sex, age, or population of some animals, or if tallies of an ISvar (such as ITOT) are used within an ISvar transition.
- In the Translocation option, use population x (rather than the last population) as the “holding population”.
- During supplementation of individuals with the Translocation option, populations are chosen in a random order to receive translocated (released) individuals, so that if there are not enough individuals in the last population to fulfill the supplements, across years and iterations each population will have an equal chance of receiving its specified supplements. However, new supplements to the last population (or the holding population, see next option) are always added after the supplements (translocations) into the prior populations.

Undocumented options are usually very special cases that won't be used by many people, or are options that are still being tested and are not yet ready for widespread use. The options are invoked by giving the appropriate codes (separated by semi-colons or spaces if more than one) in the box. A few of the current "undocumented" options are;

- **P** – obtain the mate for each female from an ISvar labeled "MATE". If MATE = -1, then a mate is selected at random from the available males. The ISvar labeled "MATE" will be the ID of the individual to be used as the mate (if available).

This option allows the user to assign specific male x female pairings. A few uses of this option might be to assign specific male-female pairings from animals in a studbook used as the starting population; or to provide a means by which another program, through Metamodel Manager, assigns the pairings.

To invoke this option, give the special option code **P**. An Individual State Variable must also be created and given the label **MATE**. (It can be any of IS1, IS2, etc.) If the ISVar of MATE is not provided, then this option is ignored, and this provides a means to have the option invoked for some scenarios but not for others in a project file.

When mates are assigned to breeding females, if the ISvar MATE is a non-zero number, and if that number matches the ID for a living animal (of either sex and of any age), then that animal is assigned as the mate for the female. The value of MATE for males is ignored. (I.e., males are assigned to females based on the value of MATE for the females, but not the reverse.) If the value of MATE is 0 (or negative), then the female is assigned a mate from the pool of available males, using the normal breeding system in *VORTEX*.

Users must be careful about interactions between mate assignments and the probability of breeding. Assignment of a mate does not guarantee that a female will breed and produce offspring. Instead, her probability of breeding each year is determined by the %breeding input parameter. Note that an easy way to enable breeding success for females assigned mates and prevent any female from breeding if she does not have a pre-assigned MATE would be to set the %breeding to be the function **=IS1*100** (in which IS1 is the ISvar labeled MATE). If females assigned mates should have a breeding rate < 100%, this can be modeled by setting the %breeding to be **=(IS1>0)*x**, in which x is replaced by the %breeding desired for those females. To set different breeding rates for females already assigned mates (say, 55) and females not yet assigned mates (say, 45), the %breeding can be set to **=(IS1=0)*45+(IS1>0)*55**.

Note that if an ISvar labeled MATE is read from a studbook in order to assign initial mates, the "ID" of the mates in the studbook will be the INDEX – i.e., the number in the sequence of individuals in the studbook – rather than the ID assigned by *VORTEX* (which would not be available within the studbook file). To have the studbook use instead a studbook name to assign initial mates, use special option **Q** – see below.

- **Q** – obtain mates from ISvar MATE (as with option **P**), but when used for reading initial mates from a studbook file, look for the male with ISvar = "studbook name", rather than

ISvar = “index” (sequence in the file). When used during the simulation years (as opposed to when reading mates from a studbook file), there is no difference between option **Q** and option **P**.

- **EP** – assign the expected number of progeny, rather than the actual number of progeny, produced in the year to the variable **PROGENY**. This can be useful in models of ex situ population management in which the number of pairs created is constrained to keep the number of progeny within the limits of the available space.
- **Ix** – read x ISvars from the studbook file specifying the starting population. Replace x with a number.
- **GDMx** – Specifies the metric of genetic distance between populations to be used when a function includes the variable **GDIST**. When x=0 (the default), **GDIST** is Nei’s standard genetic distance, D. Specify x=1 for the genetic identity (I); x = 2 for gene identity, Jxy; and x = 3 for Gst. Caution: values for **GDIST** are updated in the Census of each annual cycle of events, so you might need to put a Census event before the step at which your model needs to use current values for **GDIST**. See the [Output Summary](#) section for more information on these measures of between-population genetic distance.
- **GLx** – Specifies that calculations for genetic summary metrics should be based only on loci up through locus x. This option has no meaning if in the [Genetics](#) section it is specified that there is only 1 locus, or that the genetic tallies are based only on the first locus. Also, the first locus may be included or not, based on settings in Genetics. This option can be useful if, for example, the last few loci are used for a model of selection, but it is desired to have the genetic summaries based only on neutral loci.
- **KM** – A matrix will be provided by the user to specify the kinships among some or all of the initial individuals provided via a studbook (see [Genetics](#) section). The name and path of this file with the kinships must be the same as the studbook file, except with extension “.kin”. The file must contain coefficients of kinship (usually symbolized *f*; sometimes termed the coefficient of consanguinity), NOT coefficients of relationship (*r*). (Detailed explanation of the different measures of relatedness and methods for estimating kinship from genealogical or genotypic data are beyond the scope of this manual. Seek help from a population geneticist.) The format of the file should be a simple text file, with an initial line that lists the IDs (matching the studbook numbers) of the individuals for which kinships will be provided, followed by lines giving the kinships, as follows:

```
# Comments, to be ignored by the program, are preceded with '#'
      3    9    11   Joe   A2
# IDs can be any combination of numbers and letters,
# but cannot contain spaces, commas, or semi-colons.
# Another comment: note that delimiters can be tabs, spaces, semi-colons,
# but NOT periods, because of the confusion that causes with non-American data formats.
# The decimal delimiter used should be whatever is appropriate for the data format of the
computer (e.g., a period in the USA, but a comma in most of Europe).
3; 0.5; 0.0; 0.1; -1; 0.2
9      0.0    0.55   0.1     0.1     0.0
11 0.1  0.1 0.50 0.22 0.1
```

```

Joe    -1; 0.1; 0.22; 0.5625; 0.0
# A value of -1 is a code for a missing kinship
A2    0.2    0.0    0.1    0.0    0.5
# Only the lower, left side of this symmetrical matrix needs to be provided,
# but it does not hurt to include the full matrix.

```

The list of IDs can be given one per line, rather than all on one line (as so as to match the format of empirical kinship files used for the PMx pedigree analysis program). In this case, the lines with kinships should be preceded by a line that is just:

KINSHIPS

The initial value for the ID on each row of the kinship matrix can be omitted. I.e., the above data lines can be:

```

0.5; 0.0; 0.1; -1; 0.2
0.0    0.55    0.1    0.1    0.0
0.1  0.1 0.50 0.22 0.1
-1; 0.1; 0.22; 0.5625; 0.0
0.2    0.0    0.1    0.0    0.5

```

Another alternative format for entering kinships is a file with pairwise kinships listed one per line (rather than as a matrix). The line after the list of IDs must then contain the label “PAIRWISE” somewhere in the line. An example with the same kinships as above:

```

# Comments, to be ignored by the program, are preceded with '#'
3      9      11      Joe     A2
PAIRWISE data entry
# This file format would be more convenient than the matrix when there are only a few kinships
3; 3; 0.5
3; 9; 0.0
3; 11; 0.1
3; Joe; 0.2
# Note that missing values could be entered as -1, but the lines can also just be omitted
9; 9; 0.55
9; 11; 0.1
9; Joe; 0.1
9; A2; 0.0
11; 11; 0.50
11; Joe; 0.22
11; A2; 0.1
Joe; Joe; 0.5625
Joe; A2; 0.0
A2; A2; 0.5

```

Kinships between initial individuals that are not assigned in this matrix will be set to 0 if the individuals are founders (with “Unknown” or “Wild” parents) or set to the kinship that would normally be calculated from the pedigree for descendants. Any kinships that are provided in the matrix will replace the values that otherwise would be calculated by the program, and this will also affect the kinships of all descendants of those individuals.

Note that when the **KM** option is used, often you will want the calculations of gene diversity (expected heterozygosity) to be based on the kinships, rather than on the allele frequencies at simulated loci. To achieve that, select that option in the [Genetics](#) section.

- **UNK** – When using a studbook file to specify the starting population, omit all genes descended from founders with unknown parents (UNK x UNK) from calculations of mean kinships. If gene diversity is specified to be determined from kinships (rather than simulated allele frequencies), then it too will omit contributions from Unknown founders.

Caution: Special Options are applied to all scenarios in your project. Therefore, if you want to apply a special option to only some scenarios, you will need to remember to toggle that option on and off when you run the different scenarios. Some Special Options may be moved into the standard input sections after they are well tested and deemed to be broadly useful. And some previous Special Options already have been moved into other input sections.

Simulation Input

Navigate between the pages of input by clicking on the labels in the left-side column. It often makes sense to enter data in the sequence in which the pages are listed, but you can enter data in any order.

Adding, deleting, re-ordering, and moving among Scenarios

The Add, Delete, and Reorder buttons for Scenarios are fairly intuitive. The three methods for moving between Scenarios are to click on the << or >> buttons, use the DropDown list, or click on a Scenario label from the horizontal list. If there are many Scenarios, or they have long names, then the horizontal list will extend beyond the window and further Scenarios can be accessed from a small DropDown at the far right edge of the list.

Section Notes

On each page of input, you can (and should!) enter Notes to document the sources of data or perhaps the uncertainty about data values. Notes about input values are entered within a text box for each input section. When you save your Project, these notes are inserted into the file (with .inp extension) that lists all the input values for a scenario. They are also automatically placed into a separate text file (with extension .notes) that lists only the Notes.

Short-cuts when you have multiple populations

On the input pages described below, if you have more than one population then there will be a tool below the list of input pages that lets you copy all population-specific values from any population to all the subsequent populations. You choose if to copy only for the current page or also for all pages of input.

Quick view of Deterministic Rates

When entering input values, hitting **ctrl-D** or clicking on the “Det.” icon will pop up a window that gives the deterministic rates (r , lambda, R_0 , and generation time) for each of the populations of the current Scenario. This is a quick way to check to see if your input values will produce a positive population growth, in the absence of any stochastic variation.

The screenshot shows a Windows-style dialog box titled "Deterministic Rates". The title bar also includes the text "Scenario / Description". The main area contains a table with the following data:

| | Population1 | Population2 |
|------------------------|-------------|-------------|
| r | 0.1094 | 0.1094 |
| lambda | 1.1156 | 1.1156 |
| R_0 | 1.7229 | 1.7229 |
| Female Generation Time | 4.97 | 4.97 |
| Male Generation Time | 4.97 | 4.97 |
| Mean Generation Time | 4.97 | 4.97 |

To calculate these statistics on the deterministic population growth rates, *VORTEX* uses the same approach as the one described in Ricklefs (1982), and many other population ecology texts.

As a very brief summary of the methods ...

R_0 , the net reproductive rate, is first calculated as $\text{Sum}(L_x M_x)$, in which L_x is the age-specific survival and M_x (denoted B_x by Ricklefs, but M_x by some other authors) is the fecundity, and the summation is over all age classes (x).

An approximate (and usually pretty good) estimate of T (generation time) can be obtained by $\text{Sum}(x L_x M_x) / \text{Sum}(L_x M_x)$.

And an approximate value for r (exponential growth rate) can then be obtained as $\ln(R_0)/T$.

VORTEX, however, goes to the trouble to get a more accurate estimate by determining the value of r that solves the “Euler equation”

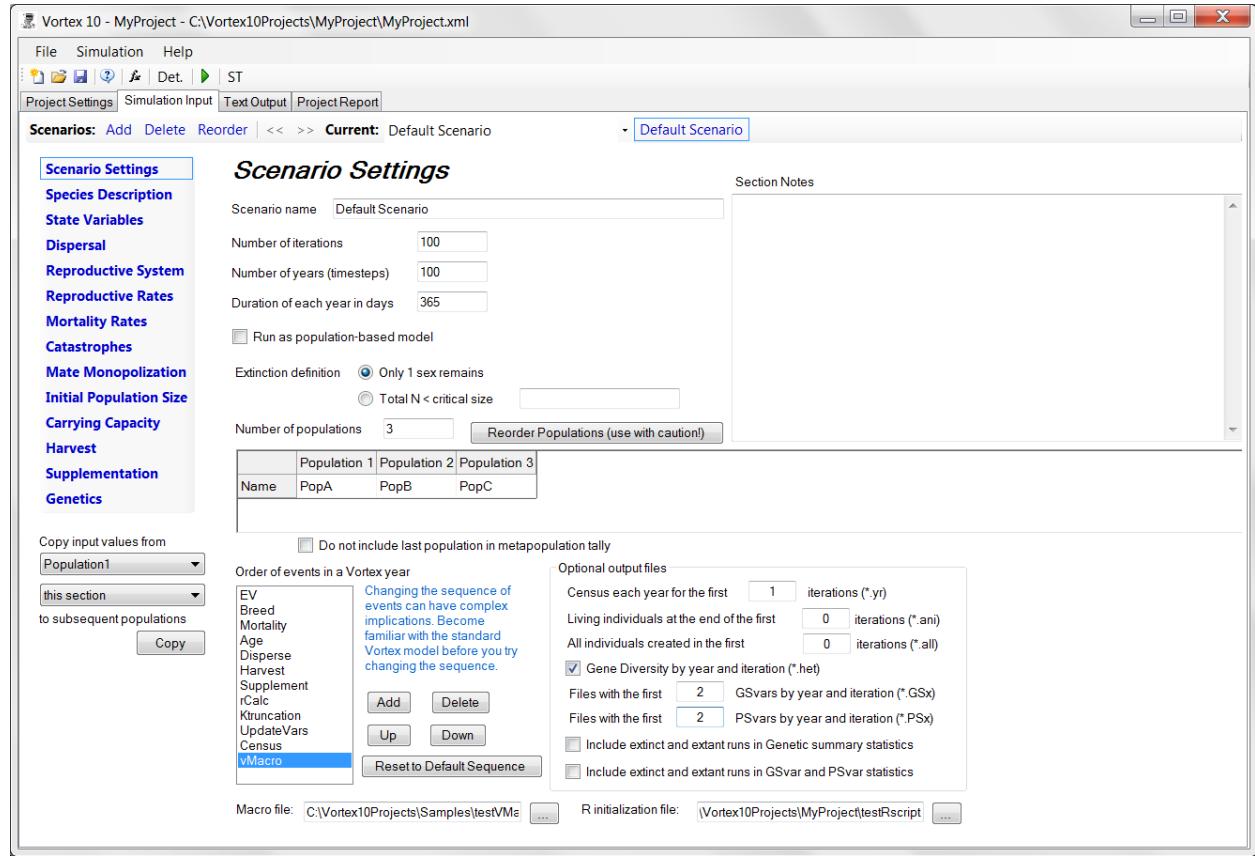
$$1 = \text{Sum}(\exp(-rx) * L_x * M_x).$$

T is then obtained as $\ln(R_0)/r$.

When *VORTEX* does deterministic calculations, it averages the impact of any catastrophes on fecundity and survival, so that the demographic rates are the means across all catastrophe and non-catastrophe years. It also adjusts for unequal sex ratio, and calculates a separate T for females and males, under the assumption that fecundity is driven by females and that the population’s total fecundity is then evenly allocated to the adult males. In these calculations, it does not consider the effects of inbreeding, density dependence, and stochastic fluctuations due to EV (all of which cause the demographic calculations to vary in unpredictable ways).

Scenario Settings

Scenario Name: The Scenario Name can be almost any label that you want. However, like the Project Name, it will be used in the filenames for output files, so do not use any characters (such as '\', '/', '*', or '?') that might be invalid within a filename.



Number of Iterations: How many times you wish to repeat the simulation, given the data that you provide in the subsequent steps? Each repetition is generally defined as a “run” or “iteration”. Because VORTEX uses a random number generator to simulate random events in the life cycle, no two iterations will be identical. Thus, to obtain a more complete picture of your simulated population, you will want to generate multiple iterations of your model.

As a first step, you may want to make sure that the simulated population is behaving in a manner that is similar to your expectations. To check this, you can limit the number of iterations to just 10 or 20. If you wish to obtain a relatively crude picture of your results, use 100 iterations. Once you are comfortable with the model and wish to obtain a more rigorous description of the simulated population’s behavior, it is not excessive to enter 500 or even 1000 iterations. Note that commas are not used when specifying larger numbers during the input process, even if your computer is set to use American data formats.

Number of years: How far into the future do you wish to project your population? The usual answer to this question is 100 to 200 years, although a shorter duration can be entered so that you

can assess the validity of your input parameters, or to examine the short-term viability of a population. If you simulate your population for just a few decades, however, you should be aware that processes controlling population dynamics might be leading the population toward extinction but, especially for long-lived species, the final extinction may not occur until a later time. By the time that the factors influencing extinction are apparent, the process may be so far along as to be almost irreversible. One of the major advantages of PVA modeling is that it can reveal the instability of a population long before it would be apparent through field observations.

Duration of each year in days: VORTEX does not necessarily require “years” to be defined as calendar years. Rather, the program operates more broadly in terms of “time cycles”. If the species you are modeling has a short generation time and life span, on the order of weeks or months—such as mice or shrews, for example—true calendar years would be an inappropriate time scale to use for modeling population dynamics. In this case, a “year” for this type of species may actually represent only one or a few months. VORTEX does not use the value that you put into this box; the box is there primarily to remind you if you choose to use a duration of the time step that is different than the one calendar year. However, this value is important if you link your VORTEX model to other simulations via the *METAMODEL MANAGER* software (see <https://scti.tools/metamodelmanager/> for more information). *METAMODEL MANAGER* uses this variable to know how to synchronize the VORTEX simulation with other simulations that might be using a different time cycle.

When specifying demographic inputs, it is vitally important that you consistently adjust input values to be appropriate to the time cycle that you are using (see Box below for an example).

Calculating input parameters when the “time cycle” is less than one year

Consider a hypothetical rodent population where the average generation time is 180 days. In order to model this population most effectively in VORTEX, the user must adjust the “time cycle” to account for this shortened generation time. In this case, we will define a VORTEX “year” as 90 days. Consequently, events whose occurrences are typically described on an annual or per-generation basis must be redefined in terms of the new definition of “year”.

For example, consider a major catastrophic flood that is thought to occur on average once every 100 years. The annual probability of occurrence, then, is 0.01. Because of the altered definition of “year”, the rodent model must define the probability that this flood will occur in any given 90-day interval. The number of 90-day time cycles in a calendar year is $T = 365 / 90 = 4.06$. Therefore,

$$\Pr(\text{flood})_{90} = \frac{\Pr(\text{flood})_{365}}{T} = \frac{0.010}{4.06} = 0.0025.$$

The same considerations must be applied to all other demographic rates, such as mortality, age of first and last breeding, etc. In addition, appropriate migration, harvesting and supplementation rates must be established relative to the revised time cycle.

Population-based modeling: The simulation can be run as a population-based model, rather than as an individual-based model. In a population-based simulation, all genetic options and modeling

(e.g., of inbreeding depression) are disabled, as is individual variation (demographic stochasticity). Population-based models will run *much* faster than do individual-based models.

VORTEX 10 has no hard-coded limit on population size. Computer RAM might limit population size if inbreeding depression is modeled. Inbreeding calculations will be very slow if $N > 10,000$. If N typically stays above about 1000 throughout the simulation, it might be more efficient (much faster, with no detectable change in results) to run the scenario as a population-based simulation. Note that inbreeding calculations are fast (and do not require much memory) if inbreeding depression is modeled as due 100% to recessive lethal alleles (and if no function uses inbreeding, **I**, or kinship, **KIN**, as a variable).

Extinction Definition: *VORTEX* gives you two methods to define “extinction” of your population. For most sexually reproducing species, ultimate biological extinction is assured whenever the population has declined to the point that it no longer has individuals of both sexes. In the first (and most common) choice, extinction is simply defined as the absence of at least one sex.

You also have the option to assess the probability of a population dropping below a user-defined threshold size – termed *quasi-extinction*. The use of quasi-extinction risk offers a useful alternative to the standard extinction risk. If you chose to have the simulation tally quasi-extinctions, you need to specify the threshold critical size below which a population is considered extinct. The simulation will, however, continue to run, as the population may grow again to a size above this threshold. Such recovery from quasi-extinction would be tallied as a recolonization event. However you define “extinction” for the purpose of tallying extinctions, in the Tables & Graphs section (see below) you will be able to assess the probability of quasi-extinction across the range of final population sizes.

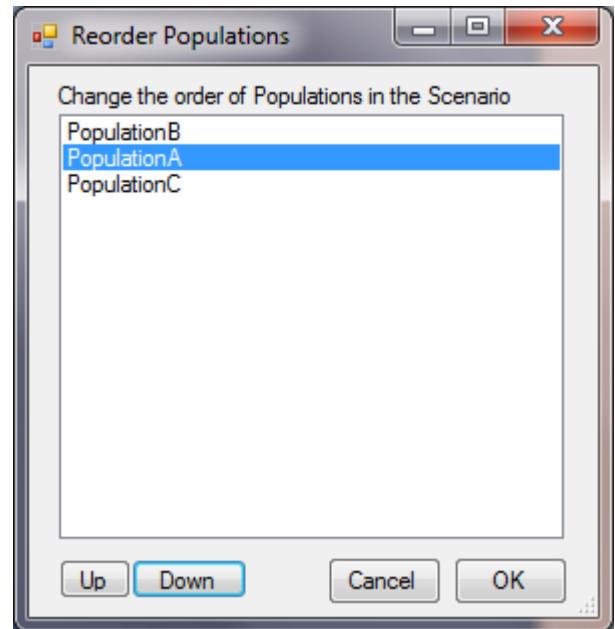
If you select the option to have extinction defined as N dropping below a critical size, you can instead of entering a number for that critical size, enter a function that defines extinction in your model. The function needs to begin with **=**. For example, if you specify the “critical size” as being **=($F < 5$)**, Vortex will define extinction as being when the number of adult females falls below 5.

Number of Populations: *VORTEX* can model a single population or a complex *metapopulation* composed of any number of populations. A metapopulation is a group of populations which, because they often occupy fragmented, discontinuous habitat, exchange individuals with varying frequency. Note that, because of the added complexities associated with metapopulations, these models will often run considerably slower than the corresponding single-population models.



The order of your Populations can later affect the ease of data entry (e.g., you can quickly copy demographic rates from a population to later populations, but not to earlier ones), or even the functioning of the model (e.g., in the [Translocation](#) option, the last population will usually be used as a holding location). Therefore, it is useful to think a minute about what order you want your populations.

Reorder Populations: You can change the order of the Populations by clicking on the “Reorder Populations” button. This feature should be used sparingly and with caution, because (as noted above) the order will affect how Translocation is implemented and also will affect how functions of Population (variable P) are interpreted. However, there are times when it is very useful to reorder the Populations. For example, if you have a Scenario with 10 populations and using Translocation, and you then want to insert a new Population (but not have it be the holding location for Translocation), you can increase the number of Populations and then move the new one up so that it is not last. When you are done moving Populations up or down in the list, hit “OK”, or hit “Cancel” if you want to reject the changes and stay with the original order.



Population labels: The names that you give to each population are used only to label output. Use any names that you want, but as with Project and Scenario names, do not use any characters (such as '\', '/', '*', or '?') that might be invalid within a filename, because the population names will be used within some filenames.

Do not include last population in metapopulation tally: This option is sometimes useful when the last population represents an external source population for dispersal into your other populations, or is the temporary holding population for a model with translocation.

Annual sequence: The sequence of events in the annual cycle can be specified to be something other than the default (EV [setting annual rates]; Breeding; Mortality; Aging; Dispersal; Harvest; Supplementation; Calculate Growth; Carrying Capacity; Update GSVars, PSVars, and ISVars; Census). Steps can be placed in any order, and steps can be repeated (e.g., there can be dispersal both before and after harvest). Updating of state variables can be done as one operation (with the order being GS, then PS, then IS), or the levels of state variables can be updated independently.

Age classes for which input values need to be specified will be determined by which steps are placed before Aging. For example, if Harvest occurs before Aging in each year, then the 0 age class animals can be harvested. If two breeding cycles occur without an Aging step intervening, then 0 age class individuals can breed. VORTEX automatically adjusts input page displays to accommodate breeding, initial N, harvest, etc. of age 0 animals if the specified sequence of events makes it possible for age 0 individuals to experience those events.

A few examples of the use of alternative order of events are:

- Harvest can be moved before Mortality if you want to be able to harvest some newborn animals. Note, however, that for this sequence to work you will also need to select the Special Option to “Delay 1st year mortality” (see above).
- Harvest can be moved before Age, but after Mortality, if you want to harvest the newborns, but only after initial mortality occurs.

- Updating of state variables can be moved up if you need the current values to be updated (with their Transition functions) before Dispersal or Harvest or Supplementation. You can also update only GSvars, PSvars, or ISvars – in any order – if you want, for example, to have PSvars contain a tally (e.g., using ITOT1) of an updated ISvar.
- You can insert additional censuses, e.g., after breeding or after dispersal, if you want to see the population counts at various points in the annual cycle. If there are additional censuses in a year, then output files will tally statistics at each census rather than only once per year.

VORTEX now allows evaluation of a VMacro to be added as an event in the annual sequence. If VMacro is added to the event list, a prompt will appear below the list where you specify the VMacro file to be evaluated. See the section on [Using VMacros](#) for more information.

R initialization file: *VORTEX* now allows you to call R scripts to modify state variables or to assign values to input variables. See the section on [Using R scripts](#) for more information about the use of R within *VORTEX*. If you are using R scripts anywhere within your Project to perform calculations, you will normally need to load the necessary R libraries at the start of the simulation, and perhaps perform some other initial calculations in R. You enter the name (usually including also the path) of the file that contains these R commands at the bottom of the Scenario Settings page. The browse button (“...”) allows you to go find your R script file. Specifying the R initialization file (e.g., as **C:/WorkingFolder/myRfile.txt**) causes *VORTEX* to execute an **RMACRO("C:/WorkingFolder/myRfile.txt")** command at the start of the simulation of the scenario.

Optional output files

There are a handful of options (previously in Special Options) for additional output files and for how summary statistics are calculated for output.

Census each year for the first x iterations: This census file (with extension .yr) can be useful if you want a detailed tally of the population year by year. The file can become large, so you need to specify for how many of the iterations you want to see such details.

Living individuals at the end of the first x iterations: This file (with extension .ani) is useful if you want to see details about each individual. The file can become large, so you need to specify for how many of the iterations you want to see such details.

All individuals created in the first x iterations: This file (with extension .all) shows the final state of every individual created in the simulation. Specify for how many iterations you want such details. Even with a listing for just one iteration, the file can be very large, so use with caution! One use of this file is to feed it into the pedigree analysis PMx (<https://scitools.pmx>) to conduct much more detailed genetic and demographic analyses on a simulated population. (Some further editing of the headers of the .all file will likely be required to prepare it as an input file for PMx or other pedigree analyses.) The file of all individuals from a simulation can also be used as the

input population for starting another Vortex simulation. (See the [Genetics](#) input section for information about using a studbook file to specify the starting population.)

Gene diversity by year and iteration: This file (with extension .GD) of gene diversity (expected heterozygosity) is useful for more detailed analysis of genetics. (Note that an output file of N for each year of each iteration is always produced.)

Files with the first x GSvars by year and iteration: Output x files with the values of GS1 through GSx each year of each iteration.

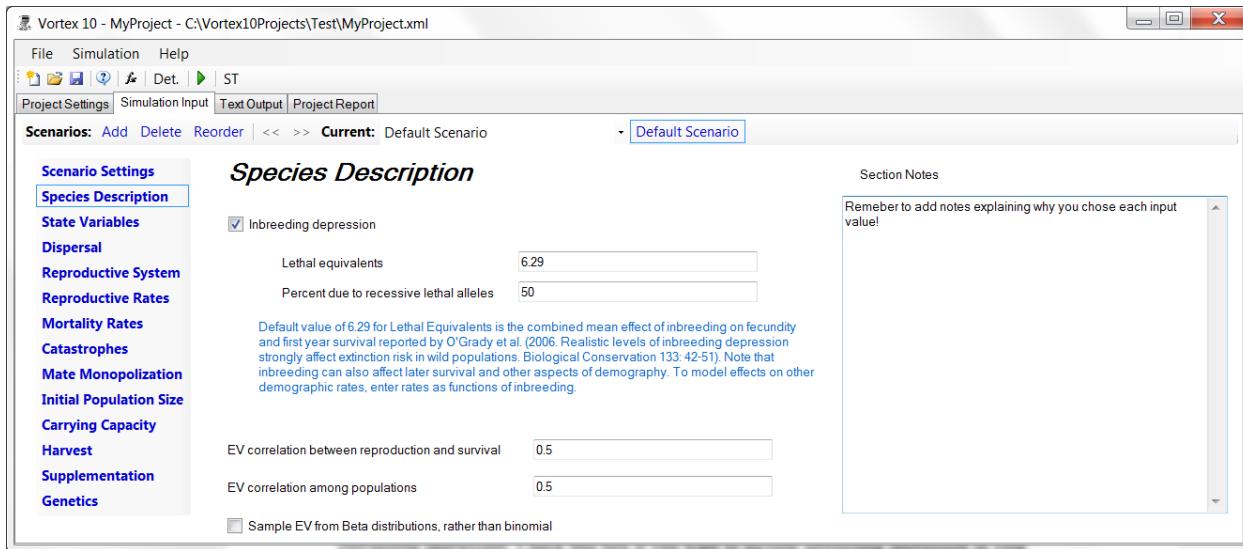
Files with the first x PSvars by year and iteration: Output x files with the values of PS1 through PSx each year of each iteration.

Include extinct and extant runs in Genetic summary statistics: Normally, population statistics reporting gene diversity, observed heterozygosity, numbers of alleles, and genetic distances include only those iterations that were not extinct (even if extinction is defined as $N >$ some critical size). This option will include all runs, even extinct ones (and even ones that have $N = 0$ and therefore genetic summary statistics of 0), in the calculation of means and SDs for genetic summary statistics. (However, this option does not affect the optional output file with allele frequencies, and genetic distances will not include runs when $N = 0$ for one of the populations.)

Include extinct and extant runs in GSvar and PSvar summary statistics: Normally, means and SDs reported for global and population state variables include only those iterations that were not extinct (even if extinction is defined in your scenario as $N >$ some critical size). This option will include all runs, even extinct ones (and even ones that have $N = 0$ and therefore state variables of 0), in the calculation of means and SDs for GDvars and PSvars.

Species Description

This input page asks a few questions about aspects of the population dynamics that will be applied across all populations of your simulation.



Inbreeding depression: Check this box if you want to include inbreeding depression in your model, as a reduction in first-year survival among inbred individuals. Although most diploid species that have been studied show depressed fitness when inbred, you may sometimes want to leave inbreeding depression out of your model so that you can compare results with and without inbreeding depression – thereby allowing you to document what impacts inbreeding depression could have on population viability.

Lethal Equivalents: This box asks you to specify the severity of inbreeding depression in your simulated population, if the population becomes inbred. Enter the impact of inbreeding on first-year survival, quantified as “lethal equivalents” per diploid individual. (See [Box](#) below.)

Percent Due to Recessive Lethals: Enter here the percent of the total genetic load (quantified by the lethal equivalents you entered into the previous box) that is due to recessive lethal alleles. The number of lethals per founder will be distributed approximately as a Poisson distribution. A plausible value – one that is consistent with data on *Drosophila* and a few other species that have been studied well – would be 50%. However, cases have been reported in which nearly all of the genetic load is due to lethals, while – in other populations – virtually none of the effects of inbreeding appears consistent with the action of recessive lethal alleles (Lacy et al. 1996). You may wish to test low and high values to see if it affects your simulations of population dynamics. (It probably won’t, because it is difficult to maintain a population for long at the very small population sizes at which effective purging of recessive lethal alleles would occur.)

Quantification of Inbreeding Depression

Inbreeding depression is the reduction in fitness commonly observed when individuals are produced by matings between genetic relatives. Inbreeding depression seems to affect most (perhaps even all) species of sexually reproducing organisms, and can cause reduction in survival (of infants, juveniles, and adults), mate acquisition, fertility, fecundity, number of progeny per litter or brood, and a variety of physiological measures related to fitness such as growth rate, disease resistance, stress resistance, metabolic efficiency, sensory acuity, and behavioral dominance (see Lacy 1997 [update] and references therein).

Although inbreeding depression can affect many components of fitness, often the overall effect can be reasonably well summarized by or combined into an effect on infant survival. For example, if inbreeding causes a 10% reduction in litter size, and then a 10% reduction in survival of those individuals born, the cumulative effect would be the same as a 19% reduction in infant survival (resulting in 81% of the yearlings which would have been produced if no inbreeding had occurred). Therefore, the primary way in which inbreeding depression is incorporated into *VORTEX* is through a reduction in first-year survival of inbred individuals. (If desired, inbreeding effects on later survival, reproduction, carrying capacity, and even dispersal can be modeled using functions of inbreeding to specify demographic rates.)

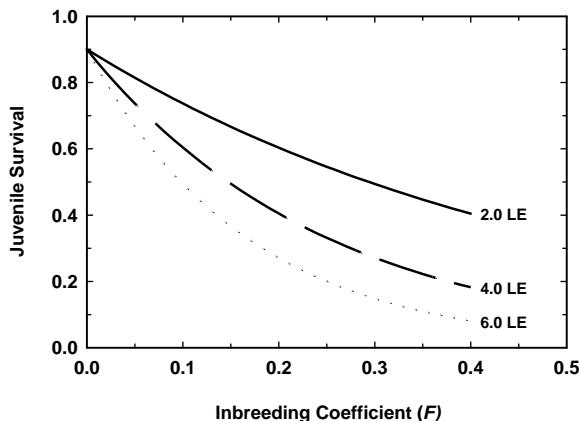
While inbreeding depression is widely known (and has been for centuries), understanding the various possible underlying mechanisms, the ways of quantifying it, and the consequences for population survival and viability is not at all simple. Inbreeding depression may result from recessive deleterious alleles (which are exposed more frequently in homozygous inbred individuals), or from a general disadvantage of homozygotes relative to heterozygotes, or from other genetic mechanisms (see Charlesworth and Charlesworth 1987; Lacy 1993b). In studies of *Drosophila* flies, it has been observed that about half of the effect of inbreeding depression on survival is due to recessive lethal alleles (Simmons and Crow 1977). The relationship between survival and inbreeding caused by the presence of recessive lethal alleles is described by an exponential decline:

$$S = S_0 e^{-bF}$$

in which S_0 is the survival of non-inbred individuals, F is the inbreeding coefficient, b is the average number of lethal alleles per haploid genome (half the number per diploid individual), and S is the resultant survival rate (Morton et al. 1956). Figure B-1 gives the expected relationship between the extent of inbreeding and juvenile survival for a series of hypothetical scenarios differing in the total number of lethal equivalents.

Even if the overall inbreeding depression is due only partly to recessive lethal alleles, the relationship between inbreeding and survival might be expected to be roughly an exponential decline of this form. By observing the relationship between survival and inbreeding, the coefficient b in the above equation can be measured. The value b is a measure of the severity of the effects of inbreeding (not in terms of how inbred the population is—as that is measured by F —but rather in terms of how much fitness is depressed for any given level of inbreeding), and it is the number of recessive lethal alleles per haploid genome that would cause the observed rate of inbreeding depression. This concept is called the number of “lethal equivalents” in the population. A population with 4.0 lethal equivalents per diploid individual ($b = 2.0$) might have 4 lethal alleles per individual, or it might have 8 alleles per individual which each cause 50% reduction in survival when homozygous, or it might have 2 lethal alleles and four 50% lethals, or any other combination of deleterious alleles which have the same total effect.

VORTEX uses this concept of lethal equivalents to quantify the severity of depression of first-year survival due to inbreeding. Thus, the user must specify how many lethal equivalents characterize the population under study. For only a few species, however, has the number of lethal equivalents been measured in careful breeding studies. Among those species that have been studied, the number of lethal equivalents per diploid ($2b$) ranges from 0 to more than 30, but it is usually in the range of 1 to 10. (Isn’t it depressing to know that you probably carry alleles which would be fatal genetic defects if you had two copies of any one of those alleles? Aren’t you glad that you are diploid?) To date, no clear patterns



have emerged to suggest that certain taxonomic, ecological, or other categories of species typically have high or low number of lethal equivalents – it seems to be largely a matter of chance whether a population is severely affected by inbreeding or not.

How does VORTEX use “lethal equivalents”?

VORTEX simulates inbreeding depression in two ways, because different genetic mechanisms of inbreeding depression can have different consequences for population viability. Recessive lethal alleles are rather efficiently removed from a population by natural selection when inbreeding occurs. As a result, many individuals may die in the early generations of inbreeding, but when they die they take their lethal alleles with them to the grave, and subsequent generations of individuals have fewer lethal alleles to cause inbreeding depression. (This process is often referred to as “purging the genetic load” of lethal alleles.) On the other hand, selection is ineffective at purging inbreeding depression when the inbreeding depression results from a general advantage of heterozygotes over all homozygotes (or, to a lesser extent, when it is caused by recessive sub-lethal alleles).

To model the effects of lethal alleles, which can be removed by selection during generations of inbreeding, VORTEX assigns to each individual at the start of a simulation some unique lethal alleles. If inbred descendants happen to receive two copies of the same lethal allele, they are killed. To model the component of inbreeding depression that is not effectively reduced by selection, VORTEX calculates the inbreeding coefficient of each individual and then applies an exponential equation like the one above (but using just a part of the total lethal equivalents) to determine how much that individual’s survival is reduced. To incorporate these two mechanisms of inbreeding depression, VORTEX needs to know (i.e., you need to tell it) how much of the overall inbreeding depression (lethal equivalents) to assign to lethal alleles vs. other genetic mechanisms. As mentioned above, for *Drosophila* flies, it has been reported that about half of the lethal equivalents are due to actual lethal alleles. Almost no other species have been studied in sufficient detail to quantify the contributions of different types of alleles to inbreeding depression, but the scant data available are not inconsistent with about half of the inbreeding effects being due to lethals in other species as well.

The initial default value for Lethal Equivalents in VORTEX 9 was 3.14, based on the mean reported for 40 captive populations of mammals. However, several authors have argued persuasively that 3.14 would under-estimate the impact of inbreeding in most wild populations, both because wild populations are subjected to stresses from which captive populations are sheltered and because the value of 3.14 was based only on effects on juvenile mortality while inbreeding impacts many other aspects of reproduction and survival as well. Some authors have argued that realistic effects of inbreeding on total fitness are modeled better with LE = 12 or even higher. Of course, you can set the value to whatever you think is appropriate for your species, and it is useful to add notes to explain your choice.

EV Correlation between reproduction and survival: Environmental variation (EV) is the annual variation in the probabilities of reproduction and survival that arise from random variation in environmental conditions. EV impacts all individuals in the population simultaneously. The sources of this environmental variation are outside the population; examples include weather, predator and prey population densities, and parasite loads. These factors can affect reproduction and survival independently or simultaneously. If this value is set to 0.0, then EV in reproduction will be independent from EV in mortality. If this value is set to 1.0, then EV in reproduction and EV in survival will be completely synchronized. As a result, good years for reproduction are also good years for survival. If this correlation is set to an intermediate value, then EV in reproduction will be partly correlated with EV in survival. (A negative correlation is also accepted, and it would indicate that good years for reproduction are bad years for survival.)

Correlating environmental variation for reproduction and survival

North America's whooping crane (*Grus americana*) shows a classic migratory pattern typical of many northern bird species. The last remaining substantial population breeds in Alberta's Wood Buffalo National Park and spends the winter at Aransas National Wildlife Refuge along the Gulf Coast of Texas. Because of this movement pattern, the environmental conditions affecting chick production are quite different from those impacting mortality during the majority of the year (Mirande et al. 1991). Consequently, we would expect EV affecting these processes to be uncorrelated when constructing a VORTEX model.

EV correlation among populations: You specify here the correlation of EV among populations (applicable, of course, only when more than one population is modeled). If this value is set to 0.0, then EV will be completely independent among populations. If this value is set to 1.0, then EV in reproduction and in survival will be completely synchronized among populations. As a result, good years for reproduction or survival in one population will lead to similarly good years in all other populations. If this correlation is set to an intermediate value, then EV will be partly correlated among populations. (This correlation cannot be less than 0, as this would cause mathematical and logical problems when there are more than 2 populations.)

Environmental variation in the metapopulation context can be considered to exist at two levels: local (population-specific) and global (acting across all populations). The total EV, when expressed as a variance rather than a standard deviation as entered by the user, is simply the sum of the EV existing at these two levels. The correlation of EV among populations that you enter, then, is simply the proportion of the total EV (when expressed as a variance) that is global in scope (i.e., common to all populations).

The Statistics of Demographic Stochasticity and Environmental Variability

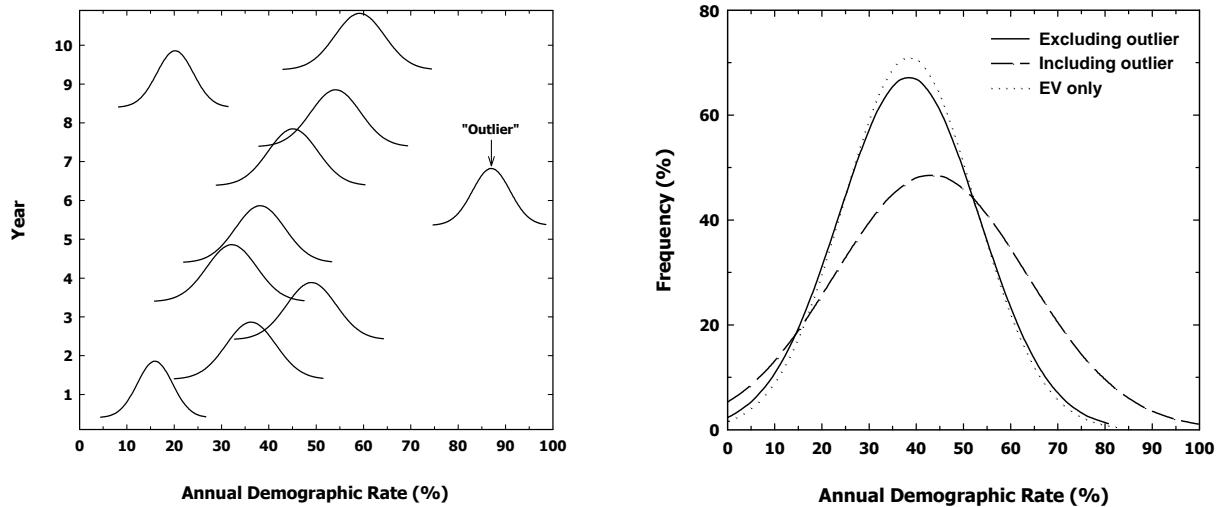
Demographic stochasticity is the random fluctuation in observed birth rate, death rate, and sex ratio of a population resulting from stochastic sampling processes, even if the probabilities of birth and death remain constant over time. This annual variation in numbers of individuals that are born, that die, and that are of a given sex can be specified from statistical theory and would be expected to follow binomial distributions.

Environmental variability is the annual fluctuation in probabilities of birth and death arising from random fluctuations in the environment (e.g., weather, abundance of prey or predators, etc.). Annual fluctuations in the probabilities of reproduction and mortality are modeled in VORTEX as binomial or, optionally, beta distributions, while environmental variation in carrying capacity is modeled as a normal distribution.

Note that the distinction between demographic stochasticity and environmental variability is a subtle one (even some professional population biologists have been confused by this!). Demographic stochasticity is the variation in an observed vital rate due to the sampling variation that is inherent because each individual (an observation) is an independent and random sample from a population with a given mean or probability. Hence, it is the variation in sample means (\bar{X}) around a fixed population mean.

Environmental variation, on the other hand, is variation (due to extrinsic factors that vary over time) in the population mean itself (i.e., the population mean is different each year).

Putting this information together, we conclude that the variation across years in the frequencies of births and deaths—both in real populations and our simulated *VORTEX* populations—will have two components: the demographic variation resulting from binomial sampling around the mean for each year, and additional fluctuations due to environmental variability. In actuality, catastrophic events (to be discussed in more detail later in the *User’s Manual*) also contribute to the overall observed variation across many years of data, but they are treated separately from standard annual environmental variability.



The left panel above shows ten years of expected values of a given demographic rate—say, juvenile mortality—in a simulated wildlife population. Each “bell-shaped” curve depicts the probability distribution we would expect from demographic stochasticity acting on that rate in that year (Actually, these little curves of demographic stochasticity would be binomial, but the normal distributions are close enough for illustration purposes). For example, the expected rate in year 1 is 15.2%. However, when the fate of each juvenile in the population is considered, it is possible that the actual rate may deviate from 15.2% solely from this sampling process. In addition, the expected mortality changes from year to year due to environmental variation, with each annual curve again reflecting the sampling variance (demographic stochasticity) expected for that year’s value. Note that these curves become tighter (the standard deviation resulting from demographic stochasticity decreases) as the means deviate from near 50%. In addition, notice that the mortality rate in year 7 is particularly high; perhaps a catastrophic event occurred in that year to produce such high mortality.

With annual rate data in hand, we can actually calculate the relative contributions that demographic stochasticity (DS) and environmental variability (EV) make to the total observed variance. Consider the example presented in the figure, the mean mortality rate calculated from these annual data is 0.387 with a standard deviation (combining effects of DS and EV) of 0.148. Note, however, that the catastrophe shown as the outlier in the dataset was not included in this calculation; if it were, the mean and standard deviation would change to 0.435 and 0.204, respectively.

If we consider the data with the outlier absent, we can calculate the standard deviation due to EV:

$$\sigma_{EV} = \sqrt{\sigma_{EV}^2} = \sqrt{\sigma_{TOT}^2 - \sigma_{DS}^2}$$

in which σ_{TOT}^2 is the total variance across the data and $\overline{\sigma_{DS}^2}$ is the mean sampling (binomial) variance across the individual rates. The binomial variance each year can be estimated as $p * (1-p) / (n-1)$, in which p is the rate observed in that year and n is the number of individuals over which that rate was calculated. The denominator is $(n-1)$, rather than the n that is more commonly seen in expressions for a binomial variance, in order to correct for a bias that occurs due to the values of p being calculated from a sample, rather than being the true probability for that year. The mean of these annual estimates of DS ($\overline{\sigma_{DS}^2}$) then provides an overall estimate of the contribution of DS to the total observed variance in the rate. In the example above, the mean binomial variance turns out to 0.0022. Therefore,

$$\begin{aligned}\sigma_{EV} &= \sqrt{\overline{\sigma_{DS}^2}} = \sqrt{0.0219 - 0.0022} \\ &= 0.140\end{aligned}$$

which is the variation across years of the mean (peak) values for each curve in the left panel of the figure. This calculation tells us that the contribution of demographic stochasticity to the total variance observed in our nine years of mortality data (remember, we removed the outlier from the analysis) is quite small—the variance attributable to environmental variability is almost 90% of the total variance in mortality. This is shown graphically in the right panel. In order for demographic stochasticity to make a significant contribution to the total observed variance, the number of individuals sampled for a given rate (n) would have to be quite small—on the order of a few tens.

The right panel of the Figure also shows the frequency distribution obtained by including the catastrophe outlier in the calculation of overall mean and variance. The inclusion of this single observation results in a significantly poorer fit to the data, as the overall distribution of values (the mean of all values in the left panel) does not look much like a normal distribution. This helps in part to illustrate why catastrophes—events that are infrequent in occurrence yet severe in population impact—are treated separately from more typical annual or seasonal fluctuations.

Finally, it is instructive to note that each of the distributions in the right panel extend beyond 0.0 and/or 1.0. As this is biologically implausible, we need to truncate these distributions in order to allow their proper use in defining probabilities. Partly for this reason, *VORTEX* usually uses a binomial distribution (which does not extend beyond 0% and 100%, but which otherwise looks much like a normal distribution) to represent EV. For ease of calculation, *VORTEX* sometimes does use a normal distribution when it is a very close approximation to the binomial, but it then truncates the normal curve symmetrically about the mean to avoid creating any bias. A Special Option allows you to ask *VORTEX* to instead use beta distributions to model EV.

The above methods are a bit complex. Because DS is usually quite small when the sample sizes (n) are at all large, a quick, somewhat generous, estimate of EV is simply the total variation in rates observed across years (treating DS as an insignificant contributor to the observed variation).

Finally, keep in mind that the *VORTEX* simulation program generates DS automatically as it determines whether each individual lives, whether it breeds, and what sex it is. Unlike some other PVA programs, you do not specify that DS should be added into the model, and you cannot exclude it (from the model or from real life). You do need to specify the magnitude of EV, however, as EV results from external processes rather than being an intrinsic and inevitable part of all population dynamics. The size of DS is a consequence of the population size; the size of EV depends on the constancy of the environment.

Sample EV from Beta distributions rather than binomial: VORTEX normally uses binomial distributions for sampling environmental variation (EV) around reproductive and mortality rates. Normal distributions will be used instead when they would be very close approximations to the binomial distributions (which happens when EV is very small). If this option is checked, however, then VORTEX will instead use Beta distributions for sampling EV in reproduction and mortality rates. Beta distributions are often used to describe uncertainty in binomial processes and, like the binomial, they are restricted to the 0 to 1 interval. An advantage of using Beta distributions is that they are continuous distributions, so the specified SD for EV can be modeled exactly, whereas the discrete nature of binomial distributions means that the user-specified SD often needs to be adjusted slightly to be modeled with a binomial. A disadvantage of using Beta distributions is that the simulations can be slower (occasionally much slower, depending on the input values) than when EV is modeled with binomial distributions. It is not at all clear which shape distribution would be more likely the best fit for EV in demographic rate, and a case could be made that if EV is the result of multiple factors in the environment (which it probably is) then EV would follow truncated normal distributions (which is what VORTEX uses when EV is small and the truncation is minor). For most PVAs, however, it seems to matter almost not at all which distribution (binomial, normal, or beta) is used for EV. In spite of debates raging among some ecologists about what shape distribution should be used for EV, it is rare to see any difference in model results when the distribution is changed from binomial to beta. This option is provided as entertainment for detail-obsessed mathematical ecologists.

State Variables



State variables add considerable flexibility and power to the VORTEX model, but they are not to be used lightly! If you are relatively new to VORTEX, you probably should skip over State Variables. Just leave this input page blank. Most population models do not need to use them, and using them effectively requires advanced knowledge of VORTEX features and especially of [Functions](#).

Any number of state variables can be created to describe characteristics of the system (“global”), each population, and each individual. These can be useful if other input values will be specified as functions of such state variables, or if a state variable tallies some metric of interest (e.g., the number of 2-year old individuals, as in PS2 below).

| Variable | Label | Initialization function | Transition function |
|----------|----------------|-------------------------|---------------------|
| GS1 | HabitatQuality | 100 | =100-Y*0.05 |
| GS2 | FadeRate | =0.2*RAND | =GS2 |

| Variable | Label | Initialization function | Transition function |
|----------|-----------|-------------------------|---------------------|
| PS1 | Area | 200 | =PS1 |
| PS2 | 2yearOlds | 25 | =TOT2 |

| Variable | Label | Initialization function | Birth function | Transition function |
|----------|-----------|-------------------------|----------------------|----------------------|
| IS1 | Color | 1 | =(DAM1+SIRE1)/2 | =IS1*(1-GS2) |
| IS2 | Is2Year | =(A=2) | 0 | =IF(A=2;1;0) |
| IS3 | OID | 0 | 0 | =MATE |
| IS4 | MateColor | 0 | =IF(MATE>0, OIS1; 0) | =IF(MATE>0, OIS1; 0) |

| Variable | Default Value |
|----------|---------------|
| A | 3 |
| N | 150 |

GSvars, PSvars, and ISvars can be referenced in functions by either the state variable number (i.e., GS1, PS3, IS7), or can be referenced by their labels. Be sure not to use a label that is the same as a built-in function (such as BROOD, PARITY, PROGENY), with the exception of a special use of ISvars to modify core properties of individuals (see below). It is also best to avoid using any punctuation (including spaces) or non-alphanumeric characters in the labels, because that would cause problems if any functions reference state variables by the labels.

Global State Variables: You can enter any number of “global state variables” describing some characteristic of the overall system. These state variables must be numeric values (or something that can be coded as a numeric value). Global state variables may describe any characteristic (e.g., climate, management strategy, or disease frequency) that you will want to be able to apply to all the populations in your model. To create one or more global state variables, click the Add

button for each variable you will be creating. A variable can be removed by clicking on its row, and then hitting the Delete button. For each variable, you then enter into the table a label, which can be any text that will help you to remember what parameter you were representing.

For each GS variable, you need to enter two functions (or constants) to define: (a) an initialization function (Init fn) – the starting value for each population at the beginning of the simulation; and (b) a transition function (Transition fn) – the change in state (if any) each year of the simulation. These functions are entered in the same way as other functions that can be used to specify demographic rates (see section on [Functions](#)).

Population State Variables: You can enter any number of “population state variables” describing some characteristic of each population. (If you have just one population, then there will be no difference between how a GS variable is used and how a PS variable is used.) These state variables must be numeric values (or something that can be coded as a numeric value).

Population state variables may describe characteristics such as measures of habitat quality or habitat suitability for the population, elevation or some other descriptor of the habitat, or perhaps some statistic (e.g., density) that you want to tally during the simulation. This option of entering population state variables is provided so that demographic rates (such as fecundity, mortality, and carrying capacity) can be specified to be functions of these state variables.

To create one or more population state variables, click the Add button for each variable you will be creating. For each variable, you then enter into the table a label, which can be any text that will help you to remember what parameter you were representing. For each PS variable, you need to enter two functions (or constants) to define: (a) an initialization function (Init fn) – the starting value for each population at the beginning of the simulation; and (b) a transition function (Transition fn) – the change in state (if any) each year of the simulation.

There is a drop-down list for selecting the population to which your PS variable functions will be applied. You can Add or Delete a PS variable only from the first population. That variable and its initial specifications are then copied over to the later populations. You then need to step through each population to adjust the Initialization and Transition functions (if necessary) to the values you intend for each population.



Make sure that you remember to set the PS variables for each population, and then re-check them all before moving on to further data input. It is easy (too easy!) to forget to set the PS variables for the populations after the first one.

Individual State (IS) Variables: VORTEX also provides the user with the option of creating any number of “Individual State” variables that define characteristics of individuals. These variables may represent any feature of the organism that can be specified or coded by a numeric value. For example, dominance status might be coded as Dominant = 1.0; Subdominant = 2.0; and Subordinate = 3.0. Or a state variable might be used to represent body condition. Or two state variables might be used to track the x and y coordinates of each individual on a landscape.

To create one or more individual state variables, click the Add button for each variable you will be creating. For each variable, you then enter into the table a label, which can be any text that

will help you to remember what parameter you were representing. For each IS variable, you need to enter three functions (or constants) to define: (a) an initialization function (Init fn) – the starting value for each individual at the beginning of the simulation; (b) a birth function (Birth fn) – the value for each newborn individual; and (c) a transition function (Transition fn) – the change in state (if any) each year of the simulation. These functions are entered in the same way as other functions that can be used to specify demographic rates (see section on [Functions](#)).

Note that the initialization function is used both for individuals at the start of the simulation and also for any individuals added in later supplementation. This provides an easy way to determine if an individual was a supplement, because the Init fn can be set to be a special code after year 0. E.g., Init fn =IF(Y=0;1;2) will assign 2 to the IS variable for all supplemented individuals.

Using IS variables to change the properties of individuals: It is possible to use ISvars to change the core properties of individuals. If an ISvar is labeled as **DAM**, **SIRE**, **SEX**, **AGE**, **ALIVE**, or **POPULATION**, then changing the value of the ISvar via a Birth, Initialization, or Transition function will change that property. For example, in some fish species, individuals begin life as males and then later transition to become females. That process can now be modeled in Vortex by creating an ISvar labeled as **SEX**. The Initialization, Birth, and Transition functions will then control the sex of the individuals. If individuals start as males and then become females at 4 years of age, the ISvar labeled **SEX** would have an Initialization function =1, Birth function =1, and Transition function =IF(A>3;0;1). Note that ‘1’ codes for male (or this could be entered in the functions as ‘M’) and 0 (or ‘F’) codes for female. If an individual changes sex, it is separated from any mate (relevant only with long-term monogamy or long-term polygyny).

In a similar way, the age of an individual can be changed via an ISvar labeled **AGE**. This can be used to create a stage-based, rather than age-based demographic model (see box below).

How to implement a stage-based model

In some species, demographic rates are determined not by the age of the individual, but rather by the stage. For example, in many reptiles and plants, fecundity and survival are determined primarily by size, and size does not always correlate with age. (Size can even decrease from one year to the next, something that true age cannot do!) Although Vortex was designed around age-based demography, it was possible to define stages via Individual State variables, and then to use the ISvars in functions that determine demographic rates. However, building a fully stage-based model in this way would be very awkward and difficult because the input and output variables are expressed in age-based terms. With the new option to control the “age” of individuals through an ISvar labeled **AGE**, it is now much easier to create a stage-based model. This can be done by removing the “Age” step in the sequence of events (so that age does not automatically increment each year), and then specifying within the Transition function for the ISvar (**AGE**) the rule(s) that define transitions between the stages. The stages would be numbered 1, 2, 3, ... (They don’t have to be sequentially numbered, but it makes demographic data entry easier if they are.) Then, in the input tabs for reproduction, mortality, etc., the rates entered would be the rates for Stages 1, 2, 3, etc., in place of the normal age-specific rates for years 1, 2, 3, etc. Interpreting output of such a model could be a little confusing, because the sequential stages will be labeled as age classes in all output files.

Changing the **DAM** or **SIRE** usually has no effect on the population model, as the genetic relationships are specified at birth, before any such ISvar would be evaluated. However, if **DAM** is used in a specification of dependency on a surviving dam (see [Reproductive System](#)), changing an individual's "dam" can make it dependent on a different individual. Changing **SIRE** might be used to designate a foster "sire" if the variable **SIRE** is used in some function. Changing the **ALIVE** property (to 0 or FALSE, for dead) is a means to remove individuals based on specific traits. However, you cannot resurrect a dead individual by setting **ALIVE** to 1 (because its record might no longer exist in the simulated population).

An IS variable with a Label of **POPULATION** can be used to move individuals between populations, based on their characteristics. Usually there would be ways to accomplish this with [Dispersal](#), using a function for the dispersal rate, but for certain kinds of movements it might be easier to use the IS variable to change the population. Each time that IS variables are updated, if there is an IS variable with name **POPULATION** and if the value of that variable is set to something other than the current population, then the individual is moved. However, if **POPULATION** is set to a number that is not a population (e.g., 0), then the individual is left in its current population. For example, a Transition fn of **=IF(I>0.20;1;0)** would place individuals with inbreeding of $F>0.20$ into population 1, and leave all others in their current population. This functionality can be used to optimize genetic management of a metapopulation, such as with **=IF(MK1<MK2;1;2)** or (if there are 4 populations) **=FIND(LOW(MK1;MK2;MK3;MK4); MK1;MK2;MK3;MK4)** to place each individual into the population where it has the lowest mean kinship. (An aside: blindly moving all individuals to the population where they have the lowest mean kinship can cause some very odd results, e.g., all individuals in population 1 can get instantly moved to population 2, leaving population 1 with no individuals and therefore not a good place to put anyone.)

Note that population statistics such as **N** might not be updated in between the updating of the IS variables for each individual, so do not use **N** (or **M**, **F**, **J**, etc.) with this option as a means to move individuals until a population reaches some threshold. **N** will be updated at times, however, when an individual in a different population is evaluated, so if you want to use the **N** for each population as it existed before the movements of individuals in the IS update started, set a PS variable to **N** and use that PS variable in your IS variable Transition fn. **MK** values are updated as each individual is moved, as those values are individual characteristics.



Do not label an IS variable as **SEX**, **AGE**, **ALIVE**, **DAM**, **SIRE**, or **POPULATION** unless you *do* want to use it to modify these properties of individuals!

Another special use of an IS variable is to give a Label of **OID** (for "other ID"). If an IS variable has that label, then the value of that IS variable specifies the ID of an individual whose IS variables are made accessible in functions as **OIS1**, **OIS2**, etc. For example, if IS1 is labeled **OID**, and it is set **=MATE** in its transition function, then OIS2 will be set to IS2 of the mate.

Default values: The table at the bottom of the page allows you to enter default values for any variables that you will be using in functions. These values will then be used when the functions are evaluated before the start of the simulation to obtain demographic rates for deterministic

calculations. This can be useful because otherwise deterministic calculations are often meaningless (as they are not “deterministic” or often even determinable). Also, the deterministic calculations of survival and fecundity schedules will be used in calculating the stable age distribution (used on the Initial Population Size page) and for some other options such as “Breed to K” in Genetic Management. Note, however, that if a reproductive rate or a mortality rate is specified as a function of age (**A**), then that dependency on age will be considered in the calculations of deterministic population growth and the stable age distribution (but not the “Breed to K” calculations each year), even if a default value for **A** is not provided by the user. Default values cannot be functions, as this could cause circular references.

It can be confusing to know what happens when you enter both an Initialization value for a GS or PS variable and enter a Default value for that variable. (What gets used to first set the value: the Default value or the Initialization?) Any Default value for a state variable is used before the iterations are started – i.e., when deterministic calculations are performed and when input functions are checked for mathematical validity. The Initialization value is then used (replacing any Default value that had been entered) at the start of each iteration. If there is no Default value given for a state variable, then the Initialization is also used for calculations that are done before the iterations start. It is not common that you would need to enter a Default value for a GS or PS variable, because usually the Initialization can be used for the same purpose. However, if an initialization function contains the variable ‘R’ (for run or iteration) – as is sometimes useful in automated sensitivity analyses – then a Default value for the state variable (or a Default value for R) can be used to ensure that the state variable is set appropriately for deterministic calculations.

State Variables extend the flexibility and power of functions for data input, and also allows you to create new output measures of interest. However, just as the use of functions can be challenging – especially if they become complex – the use of State Variables to enhance functions is best left to experienced users of Vortex. Even when using State Variables (or any functions), it is wise to first build the project without them and then add them in gradually.

An aside for those who are willing to dive deeply into using functions and user-specified variables to develop models of complex biological systems: VORTEX provides the capacity to run the population simulation simultaneously (functionally in parallel) with one or more other models that might describe the dynamics of parts of an overall system. The program METAMODEL MANAGER can manage the flow of data and the sequence of simulation steps among multiple simulation models, including VORTEX. (See Lacy et al. 2014 for a presentation of the concept of meta-models and the METAMODEL MANAGER software.) For example, several instances of VORTEX can run simultaneously, each modeling a different species that influence each other’s dynamics, as in predator-prey systems or competition. An epidemiological modeling program, OUTBREAK, can model the dynamics of an infectious disease in the population. VORTEX and OUTBREAK can be run at the same time on the same simulated population, with VORTEX simulating demographic and genetic changes and constantly informing OUTBREAK of the current census of the population, while OUTBREAK models the changes of disease state (susceptible, latent infected, infectious, recovered) and constantly informs VORTEX which individuals are in each state. The disease states can then be used to modify reproduction, survival, dispersal or other demographic rates for individuals in the meta-model. METAMODEL MANAGER and OUTBREAK are available on the SCTI website (<https://scti.tools>).

Dispersal

This section of input is accessible only if you specified (in the Scenario Settings) that your Scenario is to have more than one population. If you are modeling a metapopulation, you now need to specify a few parameters that help to define the system of dispersal of individuals among populations, and the rates of dispersal of individuals between populations.

The screenshot shows the Vortex 10 software window with the title bar "Vortex 10 - MyProject - C:\Vortex10Projects\Test\MyProject.xml". The menu bar includes File, Simulation, Help, and various icons. The toolbar has buttons for Project Settings, Simulation Input, Text Output, and Project Report. The main menu bar shows "Scenarios: Add Delete Reorder | << >> Current: Default Scenario" and a dropdown for "Default Scenario". On the left, a sidebar lists categories: Scenario Settings, Species Description, State Variables, **Dispersal** (which is selected), Reproductive System, Reproductive Rates, Mortality Rates, Catastrophes, Mate Monopolization, Initial Population Size, Carrying Capacity, Harvest, Supplementation, and Genetics. The central panel is titled "Dispersal Among Populations". It contains sections for "Dispersing classes" (Age range: Youngest 1, Oldest 5, Dispersing sex(es): Males and Females checked), "% Survival of dispersers" (50), "Dispersal modifier function (optional)" (empty), and checkboxes for "Don't allow dispersal into saturated populations" and "Use Dispersal to move a fixed number of individuals, rather than a percent". Below these are buttons for "Import Rate Matrix", "Apply multiplier of 2.0", "Export Rate Matrix", and "Fill matrix with 1.0". A note says "Enter percents of individuals in each age-sex class that disperse between each pair of populations each year; Row = Source; Column = Recipient population". A 3x3 grid table follows:

| | PopA | PopB | PopC |
|------|------|------|------|
| PopA | 95 | 2 | 3 |
| PopB | 5 | 95 | 0 |
| PopC | 5 | 5 | 90 |

On the left, under "Copy input values from", it says "PopA" and "this section" with a "Copy" button. On the right, there is a "Section Notes" pane.

Dispersing classes

Age Range – Youngest and Oldest: In these boxes, enter the youngest and oldest ages of those individuals that move between populations. If both sexes are capable of moving between populations, and the ages at which males and females disperse are different, you must decide which age(s) you use for these fields. This decision will be based largely upon how conservative you want to be about your estimation of potential risk. For example, if males begin moving among populations at 3 years of age and females at 5 years of age, entering **3** as the youngest age to disperse may underestimate the risk of population decline and/or extinction since females are allowed to move at an earlier age in the model. (It is possible to specify different rates of dispersal for the two sexes, or for different age classes, by using the Dispersal Modifier Function, described below.)

Dispersing Sex(es): Check the appropriate box(es) to specify whether males, females, or both can disperse from the natal population.

Percent Survival of Dispersers: Often, dispersal among populations occupying discrete areas of suitable habitat is dangerous. Traversing the matrix of unsuitable habitat between populations may expose an individual to additional risks of predation or lack of food, and entry into a new population may require competition with the established residents. Enter here the survival rate (as a percent) of individuals that are dispersing between populations. The dispersal mortality is imposed separately from other mortality detailed elsewhere in the program. A dispersal survival rate of **80** means that there is a 20% chance that an individual will die during the process of moving from population A to population B.

Dispersal Modifier Function: Dispersal patterns can be very complex and determined by many factors. *VORTEX* does not provide a full model of dispersal across complex landscapes, but instead models movements among discrete populations, with the user specifying the rate of movement between each pair of populations. However, this box provides you with the opportunity to customize dispersal, in perhaps very complex ways. Any function entered here will be used as a modifier of the rates to be entered later. For example, you could cause dispersal of males to be twice as high as the specified rate (and twice as high as for females), by entering **=D*(1+(S='M'))**. The parameter ‘D’ in the equation stands for the specified dispersal rate between any two populations. Such dispersal modifier functions can be used to cause dispersal to be dependent on sex, age, inbreeding, population density, and many other characteristics of the individuals and populations.

With respect to dispersal or other aspects of population dynamics, the standard *VORTEX* model may not match precisely the behavior of your species. However, *VORTEX* does provide the capability to create models that are more complex – sometimes much more complex! – than the standard *VORTEX* model. These more complex population models are built by using functions rather than constants for input values. Using such functions provides considerable flexibility, but you should use them cautiously if you are not yet fully familiar with the *VORTEX* model.

A cautionary note about the optional modifier functions and similar criteria on other input screens: Don’t make the mistake of entering a 0 if you don’t want to use that optional function, because that will often be interpreted by *VORTEX* as a function that always turns off the process (because the criterion always fails, with 0 = false). If you don’t want to use an optional function, then just leave the box blank.

Don’t allow dispersal into saturated populations: This option will stop individuals from entering a population if it is already at its carrying capacity, thereby preventing that immigration from immediately causing the population to exceed K and be truncated. If a dispersing individual had been scheduled to be moved into a population that is already at K, then the individual will move to a different population instead (if there is an unsaturated one available to receive immigrants).

Note that the behavior of the model (and of the individuals!) when dispersal is precluded because initial destinations are saturated can be complex. First, dispersal mortality is always applied to every individual that disperses or tries to do so. (But this mortality is only applied once, even if the individual moves from population to population seeking one with N < K.) Then, if the attempted destination population is at K, the individual moves on, seeking another population,

with new relative dispersal rates being the rates from the initial, attempted destination to each possible alternative destination.

This search for a new home is repeated for up to 10 times. If after shuffling among populations, seeking a place with $N < K$, the individual never finds a new home willing to accept it, it dies. There is a limit (and probably lower than 10 movements in most species!) of how long and far a dispersing individual can search for an opening before it does succumb.

An individual that fails to be admitted into a population that has $N \geq K$ can go back home, if that happens to be the new destination picked after the individual is forced to move on from a prior, saturated population (and if the original home isn't also at K). That means, for example, in a 2 population scenario, individuals that try to disperse and fail because the other population is at K will always go home again, because there is nowhere else to go. (But they do suffer the normal dispersal mortality for having tried to disperse.)

Note that if all populations are at or above K , so that all dispersing individuals die because they can't find new homes, that isn't as nasty (nor biologically unrealistic) a consequence as it might seem. That is because some individual would have to die anyways, during the K truncation if not before, because every population is at or above K . Indeed, the original source population for the disperser would have to have been above K (and not just at K) for this to happen, because otherwise after the dispersing individual leaves, its old home would now be below K , so it could then go home again.

Use Dispersal to move a fixed number of individuals, rather than a percent: With this option checked, the values in the dispersal matrix specify a fixed N of individuals that will disperse, rather than a percent. If that many individuals are not available to disperse, VORTEX will move all of those that are available.

Dispersal Rates

In the table at the bottom of this page, you enter dispersal rates to specify the probability that a given individual of the appropriate age-sex class will disperse from population A to population B in a specific year. That is, a rate of **1.00** indicates a 1% probability that an individual will migrate from population A to population B.

Dispersal rates need not be symmetric among populations; enter whatever probability you deem appropriate for each pair of populations. Enter **0** to indicate no exchange of individuals between a pair of populations. The values on the diagonal of the table – the percents of individuals that do not disperse each year – is automatically calculated by the program so that the rows will sum to 100%.

In the Dispersal Rates section are four commands that can make it easier to enter dispersal rates. “Import Rate Matrix” allows you import the table values from a semi-colon delimited text file. This file can be created in Excel or whatever software you choose. It must contain values for all cells of the table, including the labels (although the labels in the file will be ignored and will not over-write what shows on the screen). The easiest way to see the format of the rate matrix file is

to select “Export Rate Matrix”, and then look at the file that was created. With these commands, you can create a large matrix in a spreadsheet program, and then import it into *VORTEX*, and you can export rate matrices for modification or for re-use in other *VORTEX* projects. When you have only a few populations, these commands are usually not worth using. However, if you have, for example, 40 populations, you very well may want to use Excel to generate your dispersal rate matrix rather than typing in all 1560 pairwise dispersal rates!

Another command allows you to apply a multiplier to each non-diagonal cell in the table. By entering a value and hitting “Apply Multiplier of” you can shift all of the dispersal rates upwards or downwards. This makes it much easier to test a range of dispersal rates across Scenarios of your Project. For example, you might enter an initial set of rates, and then apply multipliers of **0**, **2**, and **4** in order to test no dispersal, and 2x, and 4x increases in dispersal. The fourth command (“Fill Matrix with”) in this section lets you quickly fill the non-diagonal elements of the matrix with some constant dispersal rate. This makes it easy to set up a meta-population with a uniform rate of dispersal among all pairs of populations.

VORTEX provides you with significant flexibility in defining dispersal rates among populations. For example, rates may be inversely proportional to distance, directly proportional to habitat area, or they may be defined through a more complex determining function. However, you have the task of calculating these rates for each pair of populations—*VORTEX* does not calculate them for you based on a set of internal rules.

Reproductive System

The screenshot shows the 'Reproductive System' tab selected in the 'Scenarios' dropdown. The left sidebar lists various project settings. The main panel contains fields for reproductive parameters like age of first offspring, maximum lifespan, and sex ratio, along with a checkbox for offspring dependency. A table for density-dependent reproduction is also present.

| | PopA | PopB | PopC |
|---------------------------------------|--------------------------|--------------------------|--------------------------|
| Density dependent reproduction | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| % Breeding at low density, P(0) | 50 | 50 | 50 |
| % Breeding at carrying capacity, P(K) | 25 | 25 | 25 |
| Allele parameter, A | 1 | 1 | 1 |
| Steepness parameter, B | 2 | 2 | 2 |

Monogamous, Polygamous, Hermaphroditic, Long-term Monogamy, or Long-term Polygamy: VORTEX models breeding systems as monogamous vs. polygamous vs. hermaphroditic, and short-term vs. long-term. With monogamous breeding, there must be a male for every breeding female; males may therefore become a limiting factor restricting breeding. In polygamous models, there only needs to be at least one male for all females to have an opportunity to breed. However, in a later section ([Mate Monopolization](#)) you can specify that only a subset of males have opportunities to breed. For example, you can create a polygynous system in which some males control harems of typically 5 females, while the remaining males are excluded from breeding.

If you do not choose a “Long-term” option, then VORTEX will assume that mates are randomly reshuffled each year (or each Breed event, if there are multiple Breed events per year), but not between broods in a single Breed event. If you do specify one of the “Long-term” models, then once pairs are formed, those pairs will remain together across years of the simulation until either the male or the female dies or disperses to a different population. Note that in [Genetics](#), you can specify conditions for separating a long-term pair. Demographically, it will not matter whether you choose long-term pairings or re-arrangement of pairs each year. Genetically, there may be a small effect on the rate of loss of genetic diversity from the population.

In a hermaphroditic species, every individual is both male and female. With Hermaphroditic species, an individual can potentially mate with any individual, including itself. On a later input page ([Mate Monopolization](#)) you specify the probability of selfing.

VORTEX does not fully customize the details of mating systems because of the complexities of considering a wide variety of species and their particular characteristics. More complex breeding

systems can substantially impact genetic variation, but are less likely to seriously alter the demographic performance of a population.

Age of First Reproduction for Females (and Males): VORTEX defines breeding as the time when the first offspring are born, not the age of onset of sexual maturity or the age of first conception. The program also assumes that breeding (and, for that matter, all other events) occurs at discrete intervals—usually years, but this can be described in terms of whatever you have defined as a suitable time cycle. Thus, breeding age must be entered as an integer; you cannot enter **2.5** years as the first age of breeding but must enter either **2** or **3** years. In addition, you should enter the median age of first breeding, not the earliest age ever observed since the earliest observed age may not be typical of the normal population behavior.

Maximum Age of Reproduction: Maximum ages of breeding for males and for females can be set to be less than the maximum age. This allows modeling post-reproductive life spans.

Maximum lifespan: VORTEX will kill any individual that reaches this age. Usually, however, the annual mortality rates prevent more than a small number of individuals from ever reaching the maximum age, and the variable often therefore has little effect on population dynamics.

Maximum Number of Broods per Year: VORTEX allows you to model more than one brood being produced by each female within each year. The maximum number that can be produced is asked here so that VORTEX can set up a table on next page of input, where you will specify what percent of the adult females produce each possible number of broods.

Maximum Number of Progeny per Brood: Enter the most individuals born to a given female within a brood. You can enter the maximum number that has ever been recorded—even though such an event may be quite rare—on the next input page you can then assign a low probability of occurrence to this maximum value. Note that you can combine all the broods produced by a female within each year into just one “brood”—in which case you would give a maximum number of broods as 1 and the maximum brood size as the total number born during the year. A reason to not combine broods within a year might be if you want brood size or survival to be a function of which brood (1st, 2nd, etc.) is being considered. (Such differences among early and late broods would need to be entered on the next input page as functions of BROOD.)

VORTEX needs to know the maximum possible brood size so that it can set up a table on the next input page where you will specify the probability of each possible brood size. However, you will have the option of entering a mean and standard deviation for the distribution of offspring numbers, rather than specifying the percentage of females producing each possible number of young (see below). This removes the limitation on the size of offspring numbers that can be modeled, and therefore makes it much easier to model species with high fecundities. If you will be choosing this option, then you can enter **0** (or anything at all) on this page for the maximum number of progeny per brood.

Sex Ratio at Birth: Enter here a number between 0.0 and 100.0 to represent the average percentage of newborn offspring that are male. This number is typically very near 50%,

signifying a roughly equal male:female sex ratio at birth. If relatively more or fewer males are born to a given female per year, enter the appropriate percentage (e.g., **55** for 55% males).

Make offspring dependent on their dam for x years: With this option checked, a newborn is specified to be dependent on the dam until it becomes **x** years old. This will use IS1 (*which will be created for you if it does not already exist*), giving it appropriate Initialization Fn (usually =0), Birth Fn = 0, and Transition Fn = IS1 as a place to store for each female the number of currently living, still dependent offspring. Moreover, if the dam dies, then all currently dependent offspring are killed. This options allows you to specify, for example, that only females with no dependent young can breed each year. You would do this by setting the “percent of adult females breeding” on the next input section to a function such as **=IS1<1)*50.**

Note that if you change the DAM of an individual via an ISvar (see section on [State Variables](#) above), then the offspring become dependent on that new DAM. This can be used to model dependency on the survival of some other individual (a foster-mother, grand-mother, etc.).

If you set the number of years of dependency as a negative number, then the number of dependent offspring is tracked (for a number of years that is the absolute value of the number specified), but the offspring are not killed if their dam dies. This can be useful as a way to track how many offspring are still in a natal social group, even after their dependency on the dam has ended (e.g., “helpers” in a breeding group).

Density Dependent Reproduction: Does the reproductive rate of your species change with changing population size? That is, is reproduction low when the population is small due to a difficulty in finding mates or, conversely, does reproduction drop off when the population is large (more specifically, at high density) due to limited resources or territories, intraspecific competition, crowding, stress, etc.? If so, check the box and then enter the subsequent parameters defining the nature of the density dependence.

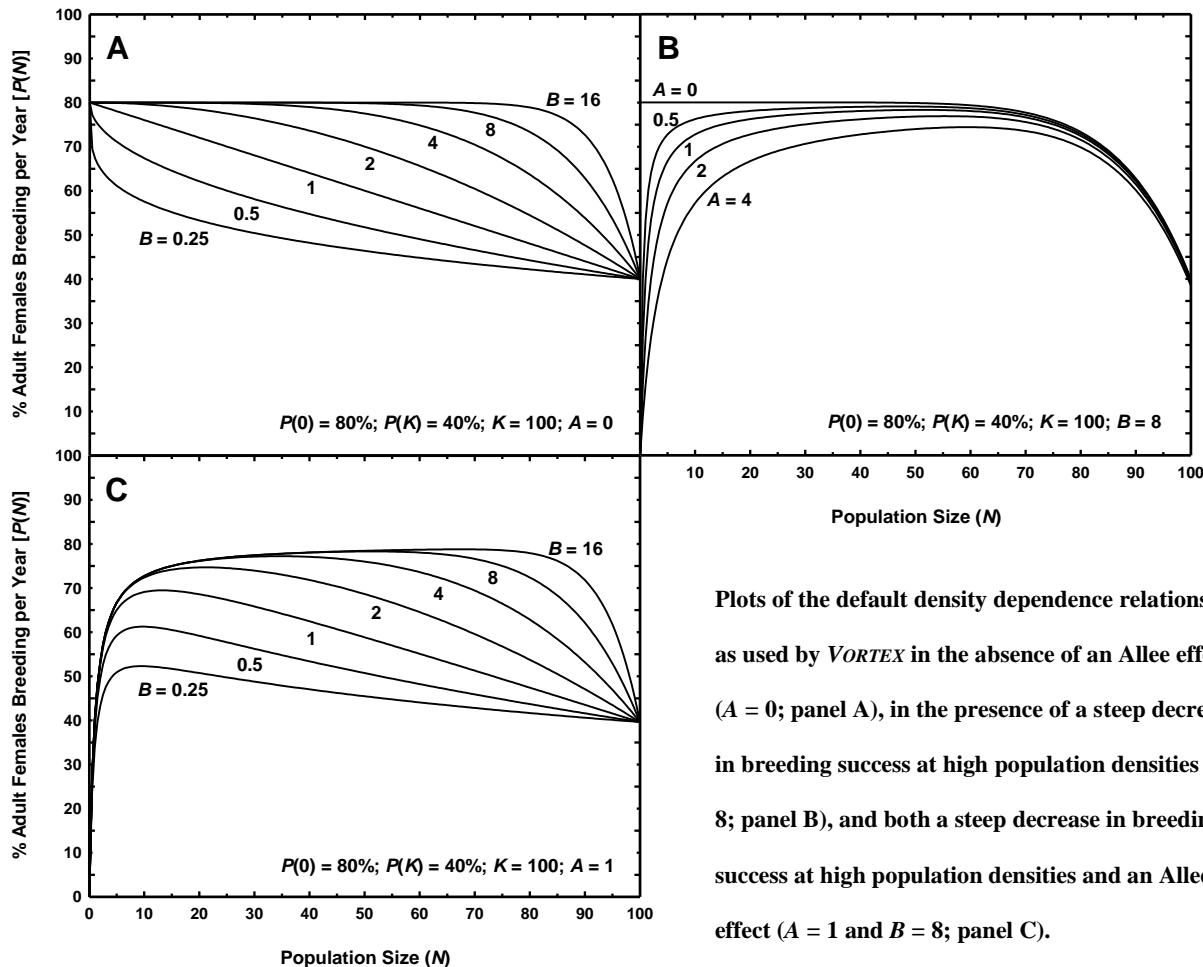
If your population’s reproduction is density dependent, you will need to model this relationship. *VORTEX* models density dependence with an equation that specifies the proportion of adult females that reproduce as a function of the total population size. Normally, the proportion of females breeding would decrease as the population size becomes large. In addition, it is possible to model an Allee effect—a decrease in the proportion of females breeding at low densities due, for example, to difficulty in finding mates.

The table that is provided to help you build a model of density dependent breeding creates a function of a form that has often been used to represent density dependence in vertebrates. If you use this table, then the function that you specify will be transferred to the first data entry box (for % breeding) on the next page. You can further edit the function there or enter any other function for density dependence. Density dependence in any other parameter in *VORTEX* (e.g., mortality or dispersal) can also be specified by functions, but the density dependence in reproduction is provided here because that is often thought to be a parameter that is particularly sensitive to population density.

The equation that *VORTEX* uses to model density dependence is:

$$P(N) = (P(0) - [(P(0) - P(K))\left(\frac{N}{K}\right)^B]) \frac{N}{N + A}$$

in which $P(N)$ is the percent of females the breed when the population size is N , $P(K)$ is the percent that breed when the population is at carrying capacity (K , to be entered later), and $P(0)$ is the percent of adult females that breed at low densities when there is no Allee effect. (Technically, $P(0)$ is the y-intercept that would result from the exponential function if parameter $A = 0$, but it will usually be approximately the breeding rate under ideal conditions.)



Plots of the default density dependence relationship as used by *VORTEX* in the absence of an Allee effect ($A = 0$; panel A), in the presence of a steep decrease in breeding success at high population densities ($B = 8$; panel B), and both a steep decrease in breeding success at high population densities and an Allee effect ($A = 1$ and $B = 8$; panel C).

It is best to derive the values of $P(0)$, $P(K)$, A , and B from a regression analysis of data on the breeding rate of your population. If these data are unavailable, but you can estimate $P(0)$ and $P(K)$, then you may want to explore several different combinations of A and B to come up with a curve that seems appropriate for your population. You could use graphics or statistical software—or even graph paper and a calculator—to construct a range of hypothetical curves, using different combinations of parameters, as was done to produce the figures above.

If you do use density dependence, it is strongly encouraged that after you have your function entered in the demographic rate, you then test that function to be sure that it represents the relationship that you wanted. If your cursor is currently on the table, hitting ctrl-F or the function icon (fx) will open the [Function Editor](#) with the density dependent function for the population of the table column on which the cursor is located.



Depending on the shape of the density dependence curve you have specified, and the mortality rates you will enter later, your population may never be able to reach the carrying capacity K . The combination of density-dependence in both reproduction and survival will determine over what range of sizes the population is expected to experience average net growth and over what range it would be expected to decline due to deaths outnumbering births.

Reproductive Rates

This section of input asks for parameter values that specify reproductive rates. Note that you decide when in the development of the next generation the “birth” is defined to occur. For mammals, you would probably use parturition as the point at which offspring are tallied. For oviparous species, however, you can start to tally offspring at egg-laying, or at hatching, or at fledging, or at any other developmental stage that makes sense to you and for which you can specify the demographic rate parameters. For amphibians, you may choose to start each animal’s life in the VORTEX model at metamorphosis. Whenever you define an individual’s life to begin, you must make sure that the first year mortality rates you specify in the next input section are appropriate for the choice you made about when to start recording offspring. For example, if you tally offspring starting at hatching, then the clutch sizes you specify on this page will be in terms of the number of hatches, and your first year mortality will be from hatching through the subsequent 12 months. If you choose to start offspring at fledging, then the clutch size will be specified in terms of the number of fledglings, and survival will be from fledging onwards.

The screenshot shows the Vortex 10 software interface with the title bar "Vortex 10 - New Project - C:\Vortex10Projects\New Project\New Project.xml". The menu bar includes File, Simulation, and Help. The toolbar has icons for Open, Save, Print, and Det. The tab bar shows Project Settings (selected), Simulation Input, Text Output, Tables and Graphs, and Project Report. A dropdown for Scenarios shows "Default Scenario". On the left, a sidebar lists various parameters: Scenario Settings, Species Description, State Variables, Dispersal, Reproductive System, **Reproductive Rates** (selected), Mortality Rates, Catastrophes, Mate Monopolization, Initial Population Size, Carrying Capacity, Harvest, Supplementation, and Genetics. Below this is a "Copy input values from" dropdown set to "Population 1" and "this section", with a "Copy" button. The main area is titled "Reproductive Rates" and contains two tables. The first table, "Distribution of broods per year (enter as percents)", has columns for Population 1 and Population 2. It shows 0 Broods for both populations. The second table, "Specify the distribution of number of offspring per female per brood", has columns for Population 1 and Population 2. It shows Mean values of 2.5 for both and Standard Dev values of 1. To the right is a "Section Notes" pane.

% Adult Females Breeding: Here you specify the mean percentage of adult females that breed in a given year (or, stated another way, the probability that a given adult female will successfully produce offspring in a given year). Data on the inter-birth interval, or the time span between successive birth events for a given female, can be useful for estimating the percentage of adult females breeding annually. A simple example: if the average length of time between successive births for adult females is 2 years, then 50% of all adult females are expected to breed in a given year (this assumes, of course, that animals can breed throughout their normal lifespan). Be

careful, however, because this estimate will be biased because you don't have data on inter-birth intervals for females that have not bred multiple times. Those poorly reproducing females should be considered when you estimate the proportion of adult females that breed in an average year.

EV in % Breeding: Environmental variation (EV) in reproduction is modeled by the user entering a standard deviation (SD) for the percent females producing litters of offspring. *VORTEX* then determines the percent breeding for a given year by sampling from a binomial distribution with the specified mean and standard deviation. Although environmental variation in birth and death rates can have a substantial impact on the viability of a population, it is often difficult to obtain the data needed to estimate EV. Long-term field studies are needed in order to determine the amount of fluctuation that occurs in the demographic rates of your population. If these data are available, the standard deviation in mean birth rate can be simply calculated using standard statistical methods. If your dataset is small but you are comfortable with making a rough quantitative estimate of the variability, you can use the technique presented below.

A common problem in estimating annual fluctuations in demographic rates is that the data might be so sparse that it is difficult or impossible to estimate the parameters on an annual basis. If this is the case, you might be forced to admit that the data are not sufficient to allow estimation of the variability around the mean values. The only alternatives are to guess at the fluctuations in reproductive and mortality rates, based on a general understanding of the natural history of the species, or to omit environmental variability from the model altogether (by entering **0** when each standard deviation is requested). In this case, you must recognize that a potentially important component of population variability and instability is being ignored in your analysis.

A difficulty with these approaches, which may add a significant bias if sample sizes are small, is that some of the year-to-year variation observed in reproductive and mortality rates is actually due to the expected demographic stochasticity resulting from random sampling of individuals, even if environmental factors do not cause fluctuations in the annual probabilities of birth and death. Refer to the Box below for methods of removing this source of variation as a means of estimating EV alone.

Distribution of broods per year: You can specify that each breeding female may have more than one brood (or clutch or litter) in each year. On the previous input page, you provided the maximum number of broods that can be produced by a female in a year. In this table, you now specify what percent of breeding females produce each number of broods. Note that this table refers to those females that are the "% adult females breeding" specified immediately above. The difference in *VORTEX* between a female that is not breeding and one that is breeding but produces 0 broods is that in the first case the female will not be paired with a male, while in the second case she will. This distinction will be important in monogamous species, because it determines whether any males are paired with non-successful females. If you do not use any functions for rates in this table, the last row will automatically be filled in so that the percents add to 100. That last row cannot be edited. If you do use a function in this table, it is then your responsibility to be sure that the rows will sum to 100. If the values sum to more than 100, the excess will be taken from the last value(s). If the rows sum to less than 100, the missing percents will be added to the last value.

A Quick and Easy Way to Estimate a Standard Deviation from Scant Data

Ideally, to estimate the standard deviation of a demographic rate across years, we would want to have many years (perhaps 10 or more) of field data. However, we often have information on just a few years, and often only the best and worst years in recent times. Fortunately, the expected range observed in a sample of n values from a normal distribution can be specified (see below, and Rohlf and Sokal 1981). To estimate the standard deviation of a distribution, the observed range (best – worst years) can be divided by the expected range. For example, across 15 years of observations on Sonoran pronghorn antelope (Hosack 1998), the mortality rate of fawns was 85% in the worst year and 55% in the best year. Dividing the observed range (30%) by the expected range for a normal distribution (3.47 SD units), provides us with an estimate of the SD of $30\% / 3.47 = 8.64\%$.

Mean ranges (in SD units) for samples of a normal distribution:

| Number of observations | Range |
|------------------------|-------|
| 2 | 1.13 |
| 3 | 1.69 |
| 4 | 2.06 |
| 5 | 2.33 |
| 6 | 2.53 |
| 7 | 2.70 |
| 8 | 2.85 |
| 9 | 2.97 |
| 10 | 3.08 |
| 15 | 3.47 |
| 20 | 3.74 |
| 25 | 3.93 |
| 30 | 4.09 |
| 40 | 4.32 |
| 50 | 4.50 |

Use Normal distribution approximation/Specify exact distribution: Previously, on the Reproductive System input page, you defined the maximum number of offspring produced per brood. Now you must specify the percentage of litters/clutches/broods produced by the breeding adult females that are of a given size. You have two options for specifying the distribution of numbers of progeny. You can use a Normal distribution approximation or you can fully specify the probabilities of each number of progeny. When you use the Normal approximation, VORTEX will randomly select a number of progeny for each breeding female by sampling from a normal distribution with the specified mean and standard deviation. To convert the random normal number to an integer (because real offspring don't come in fractional pieces, unlike the ones in the matrix models of population ecology), VORTEX uses "probabilistic rounding". The brood size is rounded up with a probability equal to the fractional part of the number and rounded down otherwise. This preserves the mean of the distribution at the value you specified. The distribution will be symmetrically truncated, if necessary, in order to prevent the specification of negative brood sizes and to prevent bias in the sampling of the distribution.

If you have data on the percent of females producing each possible number of progeny, and if the maximum number produced is not very large, then it is more accurate to enter that exact

distribution. You accomplish this by entering the percentage (i.e., a number between 0.0 and 100.0) for each specified size up to the maximum. For example, if the maximum litter size is 5 but the average litter is comprised of just 2 individuals, you would enter a much higher percentage of females producing smaller litters (say, 60% produce a litter of 2 but only 5% produce a litter of 5). If you do not use any functions for rates in this table, the last row will automatically be filled in so that the percents add to 100. That last row cannot be edited. If you do use a function in this table, it is then your responsibility to be sure that the rows will sum to 100. If the values sum to more than 100, the excess will be taken from the last value(s). If the rows sum to less than 100, the missing percents will be added to the last value.

A clever trick: If you want to use a distribution for the number of progeny that is something other than a normal (e.g., Poisson), you can do it! Just select “Use Normal distribution”, enter your function for the distribution as the mean, and then enter **0** for the SD. For example, to get a Poisson distribution with mean 2.5, you would enter for the “mean” the function **=POISSON(2.5)** and then enter **0** for SD.

You can even create a specified distribution (like the other option for entering progeny per brood) by giving a function that generates that distribution. (Explaining fully how to do this is beyond the scope of this little Box.) The advantage of doing this is, over just using the Specified Exact Distribution, is that you can then make that distribution be a function of other variables, such as age.

You can also enter a function for the mean and then also enter an $SD > 0$. This will generate the mean from the function, and then will add normally distributed variation, with the specified SD, around that mean.

Copy input values from: When modeling a metapopulation, you often will want to use the same values for most input parameters across all of the populations. On the left side of the Input pages is a tool that allows you to copy input values from any one population to all subsequent populations. You can copy only those values in the current Section of input, or copy values in all the input Sections.

A reminder: Throughout the process of data entry you should be adding notes to the Section Notes on input pages. Otherwise, there is little chance that you will later remember where you got all the input values, and there is no chance that anyone else looking at your project will know when you had solid data and when you were just playing with plausible guesses. Using good notes can also free you up to experiment with various possible values and to move ahead with constructing a model even when you don’t yet have a solidly justified number to enter for some input variable. After you get your model running, then you can go back to reexamine those input values about which you were most uncertain. (At least you can do this if you made good notes along the way to document where you had initially to put in a guess!)

Mortality Rates

In this section, you enter the age-sex specific mortalities. In the language of matrix life-table analysis (e.g., Caughley 1977; Caswell 2001), *VORTEX* defines mortality as the mortality rate q_x , or the percentage of animals alive at age x that die before reaching age $x + 1$. Enter mortality rates as a percent (between 0 and 100) for each age-sex class. Once reproductive age is reached, the annual probability of mortality remains constant over the life of the individual and is entered only once (but see the section on [Functions](#) for information on how to relax this assumption).

The screenshot shows the Vortex 10 software interface with the title bar "Vortex 10 - MyProject - C:\Vortex10Projects\Test\MyProject.xml". The menu bar includes File, Simulation, Help, and various icons. The toolbar has buttons for Project Settings, Simulation Input, Text Output, and Project Report. The main window has tabs for Scenarios (Add, Delete, Reorder), Current: Default Scenario, and Default Scenario. On the left, a sidebar lists categories: Species Description, State Variables, Dispersal, Reproductive System, Reproductive Rates, Mortality Rates (which is selected and highlighted in blue), Catastrophes, Mate Monopolization, Initial Population Size, Carrying Capacity, Harvest, Supplementation, Genetics. Under Mortality Rates, there are dropdown menus for "Copy input values from" (PopA) and "to subsequent populations" (this section). A "Copy" button is also present. The central area displays two tables for "Mortality Rates". The top table is for "Mortality of females as %":

| | PopA | PopB | PopC |
|----------------------------------|------|------|------|
| Mortality from age 0 to 1 | 50 | 50 | 50 |
| SD in 0 to 1 mortality due to EV | 10 | 10 | 10 |
| Mortality from age 1 to 2 | 10 | 10 | 10 |
| SD in 1 to 2 mortality due to EV | 3 | 3 | 3 |
| Annual mortality after age 2 | 10 | 10 | 10 |
| SD in mortality after age 2 | 3 | 3 | 3 |

The bottom table is for "Mortality of males as %":

| | PopA | PopB | PopC |
|----------------------------------|------|------|------|
| Mortality from age 0 to 1 | 50 | 50 | 50 |
| SD in 0 to 1 mortality due to EV | 10 | 10 | 10 |
| Mortality from age 1 to 2 | 10 | 10 | 10 |
| SD in 1 to 2 mortality due to EV | 3 | 3 | 3 |
| Annual mortality after age 2 | 10 | 10 | 10 |
| SD in mortality after age 2 | 3 | 3 | 3 |

A "Copy values from females" button is located between the two tables. At the bottom, there is a checkbox for "Delay 1st year mortality until all annual mortality is done (rather than in Breed)".

Mortality of Females (Males) as %: In these tables, enter the mean mortality rates for each age class, and enter also a standard deviation (SD) for each mean to describe the environmental variation (EV) in each rate.

Be aware that if you enter a standard deviation for each mean mortality rate that is at least half of the survival rate (100% - mortality rate), in occasional years the mortality rate will be set at 100% and the population will immediately go extinct. For example, if all age-specific mortality rates are 50% and the standard deviations are set at 25%, then in about 1 in 40 years the mortality rate after adjustment for EV will be 100% (since the rate will exceed the mean by 2 standard deviations about 2.5% of the time).



Delay 1st year mortality until all annual mortality is done (rather than in Breed): Normally, *VORTEX* imposes the 1st year mortality immediately after a brood is produced. This avoids having the population grow to very large size during the Breed step when a species has high fecundity and high 1st year mortality. This immediate imposition of 1st year mortality allows these early deaths to be removed before the kinship calculations are done, often thereby allowing *VORTEX* to

run much more quickly because large numbers of kinships don't need to be calculated for newborns that immediately die. The option to delay 1st year mortality is provided for cases in which you want to tally, disperse, harvest, or otherwise consider newborns before they are subjected to the 1st year mortality. Note, however, that any deaths due to inbreeding depression are imposed at birth, regardless of this option. This option to delay 1st year mortality is not visible for population-based models, because 1st year mortality is then always applied during the Mortality step, as that has no impact of the speed of the population-based simulations.

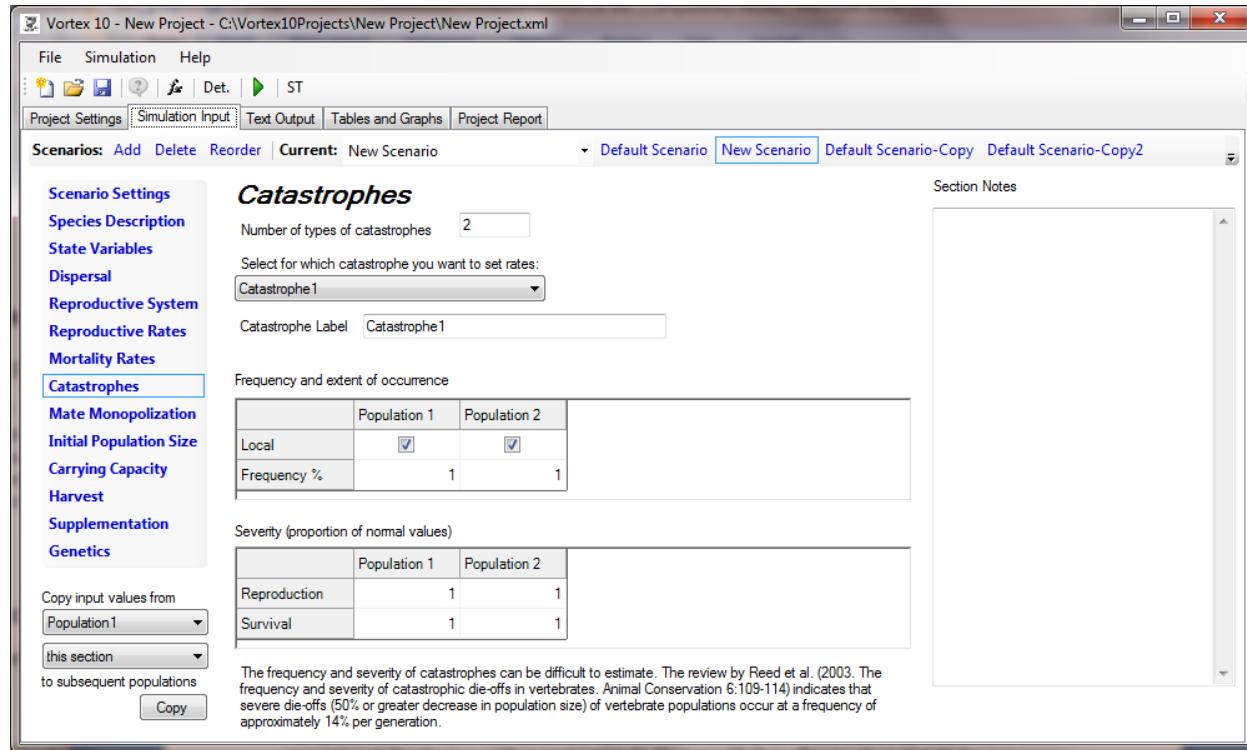
Estimating mortality rates from sightings of banded Whooping Cranes

As of 1991, the last remaining population of the whooping crane (*Grus americana*) could be tracked and censused from its breeding grounds in Wood Buffalo National Park (Alberta, Canada) to its wintering grounds along the Texas Gulf Coast at Aransas National Wildlife Refuge. During a Conservation Viability Assessment for this population, Mirande et al. (1991) estimated mortality rates for the population based on recorded sightings of banded birds. From 1976 through 1989, about 234.5 cranes were observed to hatch at Wood Buffalo NP (taking the midpoint of the possible range in those few years in which counts were imprecise), of which 172 arrived in Aransas the following winter. This yields an estimated juvenile survival rate of 73.3%. During the 14 years of close monitoring of the Wood Buffalo population, the observed variance around the mean juvenile survivorship of 0.733 was 0.047. Each year, the variance that would be expected from random binomial sampling can be estimated as $p^*(1-p)/(n-1)$, in which p is the survivorship and n is the sample size of hatchlings that year. [Note that the n-1 term in the denominator corrects for a bias that is caused by the values of p being estimated from the sample.] The mean across years of this binomial variance (mean $V = 0.013$ in these data) is then subtracted from the observed variance across years (0.047), and the difference ($V = 0.034$, or $SD = 0.184$) can be attributed to environmental variation.

Mortality after the first year can similarly be determined from either data on banded birds of known age, or from winter census reports from Aransas filed since 1938 (young of the year are distinguishable from older birds upon arrival at Aransas). Since 1938, a total of 2359 birds older than 1 year of age returned to Aransas, out of a total of 2594 birds that departed Aransas the previous spring. This yields an estimated annual mortality after the first year of 9.06%. Among the banded birds, 89.9% annual survival was observed in 386 bird-years, but band loss after several years could have accounted for some of the "mortality" recorded among banded individuals. No variation was detectable statistically among mortality rates calculated separately for each age class beyond the first year.

The observed annual variation in post-first year survival rates from 1938 to 1990 was $V = 0.00255$; the mean variation expected due to binomial sampling was $V = 0.00220$. The difference can be attributed to environmental variation in the probability of surviving, with $V = 0.00035$, or $SD = 0.019$. This value turned out to be very close to the intuitive estimate provided by workshop participants that annual fluctuations in mortality rates would be about $\pm 2\%$.

Catastrophes



Number of Types of Catastrophes: Catastrophes are extremes of environmental variation that strongly impact reproduction and/or survival. Types of catastrophes might include sudden habitat destruction, floods, forest fire, epidemic disease outbreaks, etc. Catastrophes can be significant threats to small, isolated populations. For example, disease decimated the last population of black-footed ferrets, and a hurricane killed half of remaining wild Puerto Rican parrots. It is up to you to determine what types of catastrophe, if any, may impact your population.



You may be able to identify historical catastrophes by examining birth and/or death rate data over several past years for your species of interest. If you find a demographic rate that is significantly different than that described by normal levels of variation—for example, at least 2 standard deviations from the mean value—you may use that as evidence of a catastrophic event, even if the cause was unknown.

After specifying how many types of catastrophes you want to model, you move between catastrophes by selecting from a dropdown list.

Catastrophe Label: You first enter a label for each catastrophe, as a way to remind yourself and others what events you were modeling.

Frequency and extent of occurrence: Each catastrophe is specified to be local or global in scope (this is applicable only when more than one population is modeled). You are given considerable

flexibility in how the scope of each catastrophe is specified, so it is important to read the following information carefully in order to correctly model your metapopulation.

A *global* catastrophe will occur in the same years in all populations, but the severity of effects can be entered as different or equal across populations. *Local* catastrophes occur independently among the populations. To cause a catastrophe to be *regional* in scope, affecting only a subset of the populations, you can specify that it is global, but then set the frequency to 0.0 (see below) for those populations which are not affected by that regional catastrophe.

You may also specify that a catastrophe is global for some populations, but local for others. In that case, the catastrophe happens concurrently across the populations for which it is global, but occurs independently in those populations for which it is local.

Normally, the frequency of a global catastrophe would be set to be the same in each population affected by that global catastrophe. However, you can specify different frequencies for a global catastrophe among the populations. When the catastrophe hits a population, it will also hit all other populations in which that catastrophe has at least as high a frequency of occurrence. The catastrophe will sometimes occur in the populations that have higher frequencies while not occurring in populations with lower frequencies.

Frequency %: Once the scope of the catastrophe is identified, you need to define the probability that a given catastrophe will occur in a particular year. Enter this as a percent from 0.0 to 100.0. For example, a value of **1.0** means that there is a 1% chance that this particular event will occur in any one year. Thus, a catastrophe given a frequency of occurrence of 1% means that, in a simulation lasting 100 years, this event is expected to occur one time on average per iteration.

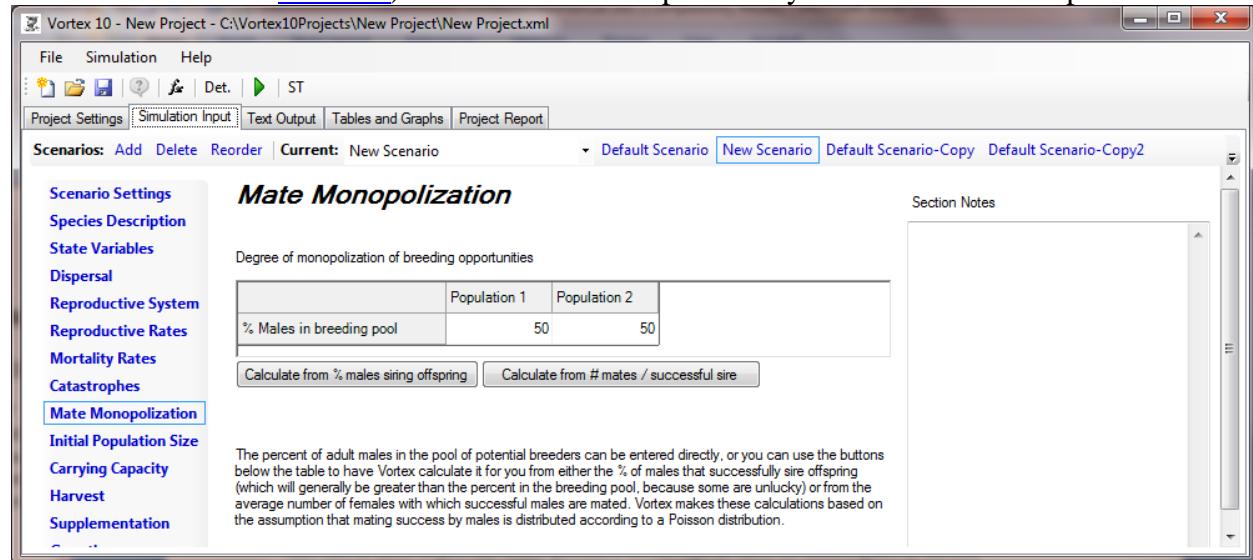
Severity (proportion of normal values): For each catastrophe, you need to define the severity with respect to reproduction (percentage of adult females breeding) and survival. The fecundity and survival rates for years in which a catastrophe occurs are obtained by multiplying those rates in a “normal”, non-catastrophe year by the specified factor. These severity factors range from 0.0 to 1.0. Entering **0.0** indicates a total loss of reproduction or survival for the population and **1.0** means that the catastrophe, when it occurs, will have no effect. For example, entering **0.75** for the severity factor with respect to reproduction means that, if 50% of adult females breed in a normal year, then only $(50\%)(0.75) = 37.5\%$ breed in a year with a catastrophe.

The frequency and severity of catastrophes are often difficult to estimate, for the very reason that catastrophes are atypical events and therefore not often observed in relatively short periods of observation. At the bottom of the input page is a reference that can be used to provide some guidance about typical rates of catastrophes in natural populations.

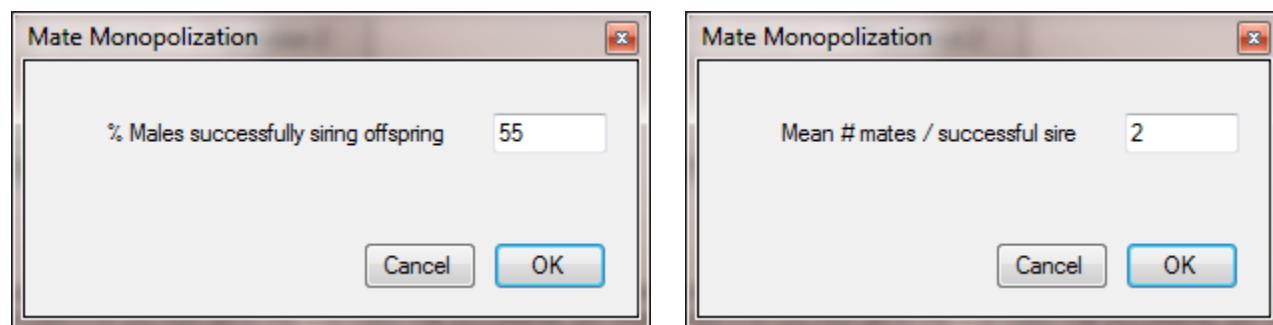
Catastrophe severity factors greater than 1.0 can be used in your model. This would result in “catastrophes” having beneficial effects on reproduction and/or survival.

Mate Monopolization

In some species, some adult males are excluded from breeding because of a lack of a breeding territory, subordinate status, physiological limitations, or otherwise. *VORTEX* will randomly sample males from a “breeding pool” (unless you otherwise constrain which males can breed via a function here or in [Genetics](#)). You enter here the probability that a male can compete for mates.



The Mate Monopolization input page directly prompts for the one measure of the degree of polygyny that is used directly by Vortex. Buttons provide access to simple pop-up utilities to calculate the % of males in the breeding pool from either the % siring offspring or the #mates / breeding male (as in the images below). These calculations will assume that the distribution of mates per male in the breeding pool is Poisson. Functions cannot be entered for these rates, because the %Males in the breeding pool is calculated from these values at the time of input data entry, so that parameters entered into a function would not yet be available for the calculation.

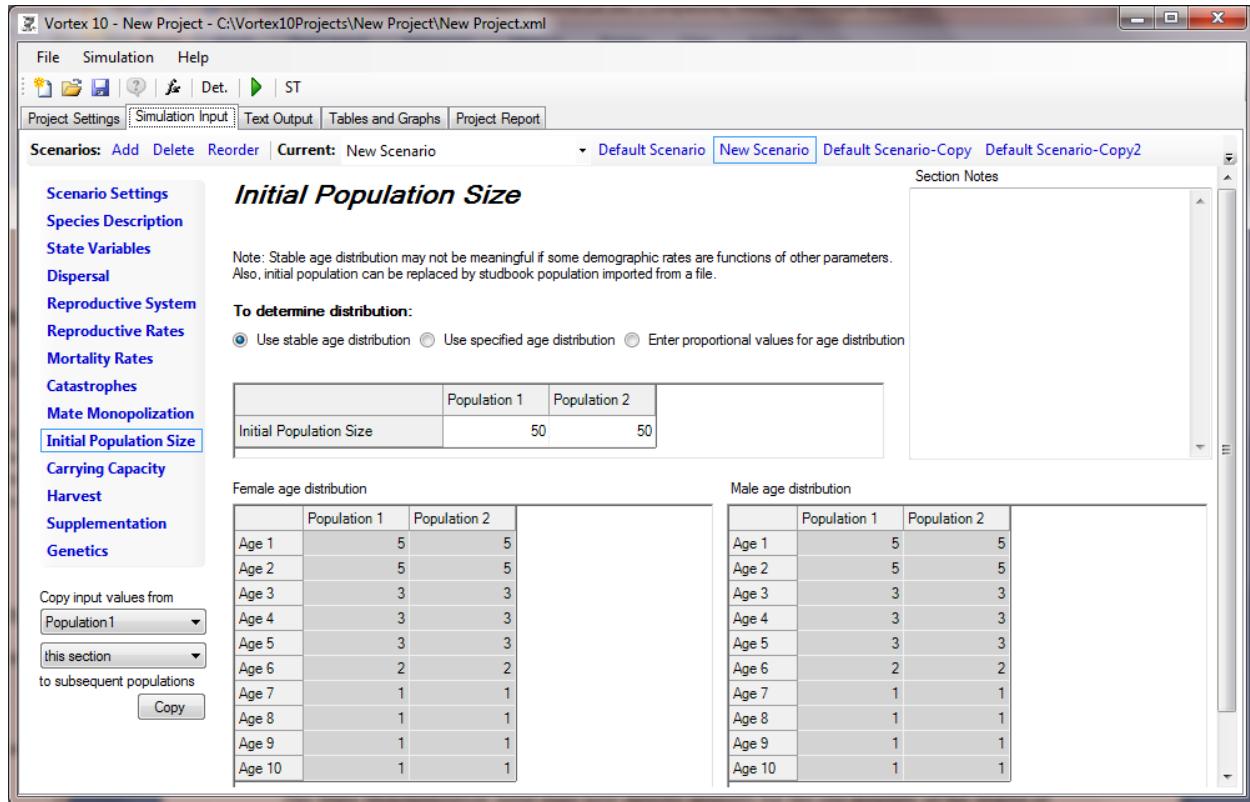


If you earlier specified that the species has a monogamous breeding system, you are asked to specify only the percentage of the total adult male population that makes up the pool of available breeders (and each male can breed with only one female each year), and these alternative ways to specify mate monopolization are not available (nor would they be meaningful).

Remember that not all males within this pool may breed in a given year, depending on the number of adult females that are successful breeders.

Initial Population Size

In this section of input, you specify the number of individuals at the start of your simulation, and the age initial structure. There are three options for how to enter the data.



If you choose “Use stable age distribution”, then you will enter the initial N and *VORTEX* will allocate them to the age-sex classes according to the expected age distribution calculated from the birth and death rates.

If you choose, “Use specified age distribution”, then you enter the number in each age-sex class and *VORTEX* totals these to fill in the Initial Population Size table.

The initial age structure of the population can instead be specified by a proportional distribution with a total initial N. With this option, you need to enter both the total N and numbers for the age-sex classes. *VORTEX* will then allocate the initial N for you according to the specified proportional distribution. The values you enter for the age-sex classes do not need to total to 1 or to N or to any specific number. *VORTEX* will use them as a relative distribution.

One way in which you can use a combination of these options is to first enter a specified age distribution. Then change the option to a proportional distribution. You can then create additional scenarios that keep that same proportional distribution even as you change the total N and other input values.

Sometimes the stable age distribution cannot be calculated if you use functions for some of your demographic rates – because in such cases the age distribution depends on the values of the variable parameters in your functions. There are two ways in which you can get a reasonable starting age distribution in such cases. You can create your scenario first with specified mean values for the demographic rates into which you will want to enter functions. Then go to the Initial Population Size page, select Stable Age Distribution, enter a total N to generate that distribution. Then change the selection to be the Proportional Age Distribution. This distribution is then locked in for the scenario even as you enter the functions that you need for various demographic rates.

Another option is to enter your functions for rates on various input pages. Then go to the State Variables input page and enter Default values for the parameters in your functions, so that the default values will represent the mean or otherwise appropriate values for the deterministic calculations that will be used to obtain the Stable Age Distribution. An advantage of this method is that then the Deterministic results displayed by Vortex will be meaningful.

Carrying Capacity

The screenshot shows the Vortex 10 software window with the title "Vortex 10 - BigCat-v10 - C:\Vortex10Projects\Test\BigCat-v10.xml". The menu bar includes File, Simulation, Help, and various icons. The toolbar has buttons for Project Settings, Simulation Input, Text Output, Tables and Graphs, and Project Report. The Scenarios dropdown shows "Baseline" as the current scenario. On the left, a sidebar lists various demographic parameters: Scenario Settings, Species Description, State Variables, Dispersal, Reproductive System, Reproductive Rates, Mortality Rates, Catastrophes, Mate Monopolization, Initial Population Size, Carrying Capacity (selected), Harvest, Supplementation, and Genetics. The main panel displays the "Carrying Capacity" settings for "Pop1". It includes fields for "Carrying Capacity (K)" (set to 100) and "SD in K due to EV" (set to 0). Below this, a note explains that EV in K can be a tricky concept. A second section shows "Future change in K?" with fields for "Over how many years?" (set to 5) and "% Annual increase or decrease" (set to -10). At the bottom, there are checkboxes for implementing K based on other variables, during K truncation, and prioritizing K truncation, along with associated cautions.

Carrying Capacity (K): The carrying capacity describes the upper limit for the size of your simulated population within a given habitat. VORTEX implements carrying capacity as a probabilistic truncation across all age classes when K is exceeded at the end of the year (or whenever in the annual sequence you have K truncation). For example, if $N = 1.25 * K$, then additional mortality will be imposed such that the survival probability for any individual during the K truncation is $1 / 1.25 (= 0.80)$, so that after the truncation the expected population size is K . You can implement carrying capacity in other ways, by specifying that some demographic rates (e.g., reproduction or mortality) are functions of N , but then it is your responsibility to determine what density dependent functions will achieve the K that you want.

SD in K Due to EV : If you think that the habitat carrying capacity varies over time due to environmental variation (EV), you can enter a standard deviation (SD) here to account for this variability. For example, the habitat might support different population sizes in different years due to changing conditions such as rainfall or food resources. The standard deviation should be entered as a number of individuals, not as a percentage of K ; for example, if $K = 2000$ with a standard deviation of 10%, then enter **200**.



Be careful! If you enter a standard deviation for the carrying capacity that is greater than $K/3$, then the value for K could drop to zero during your simulation – resulting in an immediate extinction event.

Trend in K ? VORTEX allows you to simulate changes in the carrying capacity. Such changes may be positive or negative and result from human activities such as resource utilization or corrective

management strategies, or from intrinsic ecological processes such as forest succession. To include a trend in carrying capacity, check the box. Then specify the time period during which the trend will occur, and the annual rate of change in K . The trend will begin in year 1 and continue for the specified duration. The program will model a liner trend over this time period. More complex patterns of K can be specified by entering a function for Carrying Capacity.

Implement K based on a limit on some population variable other than N: As a new option in VORTEX 10, K can be specified to be a criterion other than a limiting N . E.g., K can be a limiting number of females, or of adults, or of individuals with $IS1=1$, or of some function of variables. The proportion by which K exceeds the limiting value will determine the proportion of individuals (across all age and sex classes) that will be removed. For example, if breeding sites are limiting, with a maximum of 50, you could enter $K = 50$ and then specify to “Implement K as a limit ...” with the Population variable to be tested being **=F** (the number of adult females). However, in such a case it might be better to more explicitly build the limitation of nest sites into a density dependence function for % breeding on the Reproductive Rates page.



Caution! If you specify that K is a limitation on some population tally other than total N , then be aware that if you use K as a variable in any function (e.g., some density dependent function), then the K used in your other functions will be the one entered on this screen.

During K truncation, remove only individuals meeting criteria: Another option allows you to specify that only certain classes of individuals will be removed to bring the population back down to K . For example, you could remove only juveniles or only inbred individuals. If this criterion evaluates to 1 (or more) for an individual, then the individual is in the pool that is subject to removal. If it evaluates to a number less than 1, then that is the probability that the individual will be available for removal. This option is not available in population-based models.

Prioritize K truncation based on ISvar: Another option allows you to use an Individual State variable to specify the order in which individuals are removed when K is exceeded. This option requires the user to create an ISvar that determines which individuals will be removed first. For example, K truncation can be applied first to the oldest individuals, or those with the highest inbreeding, or those with the highest Mean Kinship. The order of removing individuals will be from the lowest value of the specified ISvar (first) to the highest. When using this option, it is important to use a sequence of events for the annual cycle that will ensure that the ISvars are updated when they are used for determining the order of K truncate individuals. Usually, this means that an ISUpdate event will need to be inserted just before Ktruncation. (See section on [Scenario Settings](#).) If a negative is placed before the ISvar code in the box, then the order will be dynamic – meaning that the ISvar will be updated again with its Transition function after each individual is removed and the list then re-sorted. In this way, the order of removals can be set, for example, to be determined by a Mean Kinship that is updated after each individual is removed from the population. (Note that to achieve removals of the individuals with highest MK, the ISvar would need to be set to “**= -MK**”, because removals are from the lowest to highest.)

What Exactly *Is* Carrying Capacity, Anyway?

“Carrying capacity—rarely in the field of resource management has a term been so frequently used to the confusion of so many.” (MacNab 1985)

The definition and use of the concept of carrying capacity is one of the more tricky issues in population viability analysis (and, for that matter, in much of population ecology). Pick up any number of textbooks on ecology or wildlife management and you are likely to find that each one presents a slightly different formal definition of carrying capacity. In fact, some authors (e.g., Caughley 1977) choose not to use the term altogether in their presentation of the mathematics of population growth. In the context of wildlife management, the habitat carrying capacity for a particular population can be defined as the maximum number of individuals that environment can sustain over time in the absence of unnatural disturbances, and without inducing harmful trends in the abundance of the resources required by that population.

We can gain more insight into this concept by considering the familiar (and admittedly simplistic) logistic equation for population growth:

$$\frac{dN}{dt} = rN \left[\frac{K - N}{K} \right]$$

where r is the intrinsic rate of population increase, N is population size, and K is carrying capacity.

Mathematically, K can be thought of as the population size limit at which the rate of growth dN/dt is equal to zero. Some ecologists define K as a ratio of the total rate of food production in the habitat (P) to the per capita rate of food consumption necessary for survival (M). Since a population of size N must consume food resources at a rate of NM just to stay alive, there are $P - NM$ resources available for the production of new individuals. If NM exceeds P , then resources available for reproduction become negative and the population must decrease in size.

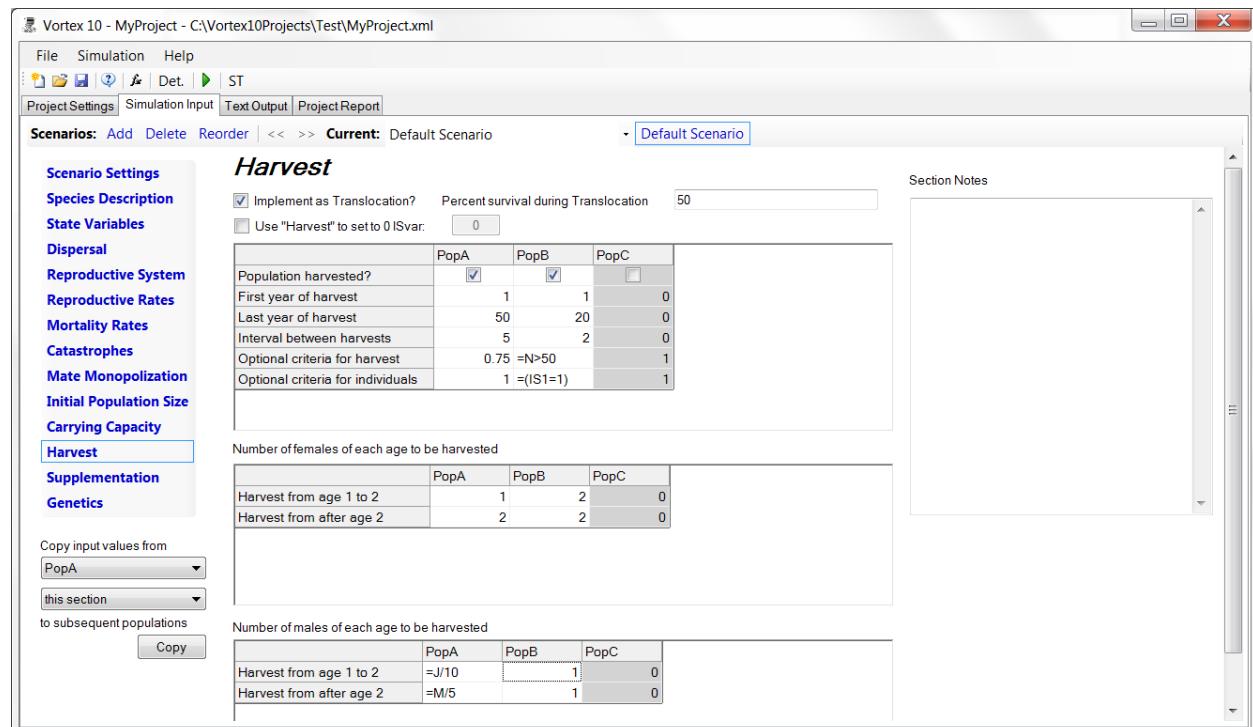
When N is very small—for example, when a population is re-established in its native habitat—the potential growth rate is very close to exponential. If the population exceeds its carrying capacity, the number of individuals will be reduced until $N \leq K$. The carrying capacity, then, can be thought of as representing a stable equilibrium population size. Many population ecologists describe the gradual approach towards this equilibrium in terms of *damped oscillations* in population growth.

Empirically, one could estimate the habitat carrying capacity for a given animal species by calculating the total food supply appropriate for that species that is available in the habitat, and dividing that value by the rate of that species’ consumption of its available food supply (for a detailed discussion of this technique, see Hobbs and Swift (1985)). For example, Petit and Pors (1996) estimated population sizes, flower availability and nectar output for each of three species of columnar cacti on Curaçao. Carrying capacities for the two species of nectar-feeding bats dependent on these cacti could then be estimated based on the daily availability of mature flowers and the field energy requirements of the bats, with additional energy requirements associated with pregnancy and lactation taken into account.

If detailed data such as these are unavailable, a rough estimate of habitat carrying capacity can be generated using long-term data on population size. If the size of the population of interest appears to be relatively constant over the period of observation (and in the absence of significant human impact), one can fairly safely assume that the population is at or near its carrying capacity. If this equilibrium is observed in the presence of major human influences, such as a strong hunting pressure, then historical data could be consulted to determine if this stable size is indeed natural or purely artificial. One could also calculate K for a given habitat using population density data from undisturbed habitats elsewhere in the species’ range.

Harvest

In this section, *VORTEX* gives you the option of removing individuals during a simulation. Harvest can mimic hunting, culling, research-related removals, removal of young individuals for translocation programs, etc.



Implement as Translocation? With this option, the last population is used as a “holding” location. (A [Special Option](#) allows you to set the holding population to be some other population.) Any individuals harvested from other populations are moved into that holding population. (Any individuals harvested from the holding population are killed.) During the movement of individuals into the holding population, a survival rate can be entered to model the loss of individuals during the translocation. This survival rate imposes an additional mortality on top of the normal annual mortality for each age class.

If the option to Implement as a Translocation is set on either the Harvest or the Supplementation page, it will be set automatically to be the same on the other one of these two pages.

Use “Harvest” to set to 0 ISvar x: During harvest, instead of killing individuals that are “harvested”, their ISx variable can be set to 0. If x is not specified, it is assumed to be IS1. This option can be used, for example, to contracept a specific number of animals each year (assuming that you also set the probability of breeding to be a function of IS1).

Check the *Population Harvested?* box to request a regular harvest of individuals.

The harvest can begin and end at any time during the stipulated length of the simulation. Enter the *First Year of Harvest* and the *Last Year of Harvest*. For example, if you wish to begin harvesting in year 10 and end in year 25, enter **10** and **25** for these two questions, respectively. If you wish to harvest every year within the specified time frame, enter **1** for the *Interval Between Harvests*. If you wish to harvest animals every other year, enter **2**. As another example, if the first year of harvest is 10, the final year is 50 and the interval is 10 years, then harvesting will take place in years 10, 20, 30, 40 and 50.

Optional Criterion for Harvest: You can specify here some criterion that will restrict harvesting to occur only if the population status meets certain conditions. You enter this as a function. For example, if you enter **= $(N/K)>0.8$** then harvests will only occur if the ratio of the population size to the varying capacity is at least 0.8 (and if it is a harvest year as defined above). Leave the table cell entry as **1** if you do not want to provide any harvest threshold criteria. The optional criteria for harvest allow you to specify conditions under which a population will be harvested. These criteria would be entered as a function of population properties (e.g., N, Y, or population state variables), which evaluates to 0 to prevent a population from being harvested that year, evaluates to 1 (or more) to allow harvest, and evaluates to a number between 0 and 1 to give that probability of a population being harvested.

Optional criteria for individuals: A new option was added to allow criteria to be specified that constrain which individuals are available for harvest. These criteria would be entered as a function of individual properties (e.g., age, sex, or individual state variables), which evaluates to 0 to prevent an individual from being harvested, evaluates to 1 (or more) to allow harvest, and evaluates to a number between 0 and 1 to give that probability of an individual being available for harvest.

Female (Male) Ages being Harvested: Enter the number of females and/or males that you will harvest at each time interval defined for each age class through adults. Enter **0** for no individuals to be harvested in a given age class. By default, VORTEX will conduct the harvest after newborn animals have aged (to become 1 year old) in the annual cycle, so the youngest individual that can be harvested is one year old. If you have changed the sequence of events (on the Scenario Settings page) to put a Harvest in between Breed and Age, then age class 0 individuals will also be available for harvest in the table.

If the number to be harvested is a function that evaluates to a non-integral number, VORTEX will use probabilistic rounding (see PROUND in the section on [Functions](#)) so that the mean number harvested is not biased.

If the program attempts to harvest individuals from an age class and finds an insufficient number of individuals, the simulation will continue without the harvest of those individuals determined not to exist. VORTEX will then report at the end of the simulation (in the text output file) that some of the attempted harvests could not be carried out.

Supplementation

You also have the option of adding individuals to each population. This option can simulate supplementation through, for example, a translocation or releases from a captive breeding program. As with the harvest option, supplemental individuals can be added at any time and interval within the specified time frame for the simulation.



VORTEX assumes that the individuals that are being added to the recipient population are unrelated to both each other and to any other individual in the recipient population. Consequently, supplementation is a means of increasing genetic diversity as well as total numbers of individuals within a population.

The screenshot shows the 'Supplementation' tab in the Vortex 10 software. The interface includes a left sidebar with navigation links like Scenario Settings, Species Description, State Variables, Dispersal, Reproductive System, Reproductive Rates, Mortality Rates, Catastrophes, Mate Monopolization, Initial Population Size, Carrying Capacity, Harvest, Supplementation (which is selected), and Genetics. The main panel has sections for 'Supplementation' settings (checkbox for 'Implement as Translocation?' with an 'if' condition =Y>10, survival probability 80%, release criteria =l<0.10), tables for 'Number of females of each age to be supplemented' and 'Number of males of each age to be supplemented' across three populations, and a 'Section Notes' pane on the right.

Implement as a Translocation? Under the option to Implement as a Translocation, the last population is treated as a holding population for transferring individuals among the other populations. (A [Special Option](#) allows you to set some other population to be the holding population.) Any individuals supplemented to populations other than the holding population will be taken from the holding population. If not enough individuals are available to meet the specified number of supplements, all those that are available will be moved. Any supplements into the holding population will be new individuals, as is the case for all supplements when the translocation option is not selected. The “if:” box allows you to enter conditions (as function or number representing a probability) under which the individuals supplemented into populations will be treated as translocations (taken from the holding population), rather than supplemented as new individuals added to the model.

Percent survival after Translocation: This data entry box specifies the survival of translocated individuals after release into the recipient population. This mortality during Translocation is imposed in addition to any normal mortality, and in addition to the mortality that occurs during the Harvest of individuals for a translocation. Note that this is a change from earlier versions of VORTEX that imposed translocation mortality only at the time of harvest, even though that parameter was before available on both the Harvest and the Supplementation tabs.

Optional criteria for individuals to be released: You can specify that individuals must meet criteria to be available for release from the last population during a Translocation. For example, you can specify that only individuals with low inbreeding will be released, or only those with an IS variable that is set to the recipient population number. In this way, you can (although sometimes with difficulty!) control which individuals are translocated into which populations.

Population Supplemented? If you want to supplement your population(s), check this box. You must then provide the values on the subsequent lines to define the nature of the supplementation.

First (Last) Year of Supplement: The supplementation can begin and end at any time during the stipulated length of the simulation. Enter the years in which you wish to begin and end supplementation.

Interval Between Supplements: If you wish to supplement every year within the specified time frame, enter **1**. If you wish to supplement every other year, enter **2**.

Optional Criteria for Supplements: You can specify here some criteria that will restrict supplementation to occur only if the population status meets certain conditions. You enter this as a function. For example, if you enter $=\text{(N/K)}<0.25$ then supplements will be added only if the ratio of the population size to the varying capacity is at less than 0.25 (and if it is a supplementation year as defined above). Leave the table cell entry as **1** if you do not want to provide any criterion for supplementing.

Female (Male) Ages being Supplemented: Enter the number of females and/or males that you will add at each time interval defined for each age class through adults. Enter **0** for no individuals to be supplemented in a given age class. These parameters differ slightly from the parameters defining harvesting in that the last age class corresponds to the first year of adulthood instead of the aggregate adult stage. This difference results from the fact that while harvesting selects any adult individual regardless of age, VORTEX must assign a specific age class to each adult that is being added to the recipient population. The age of adults added to the population is always equal to the age at which breeding commences.

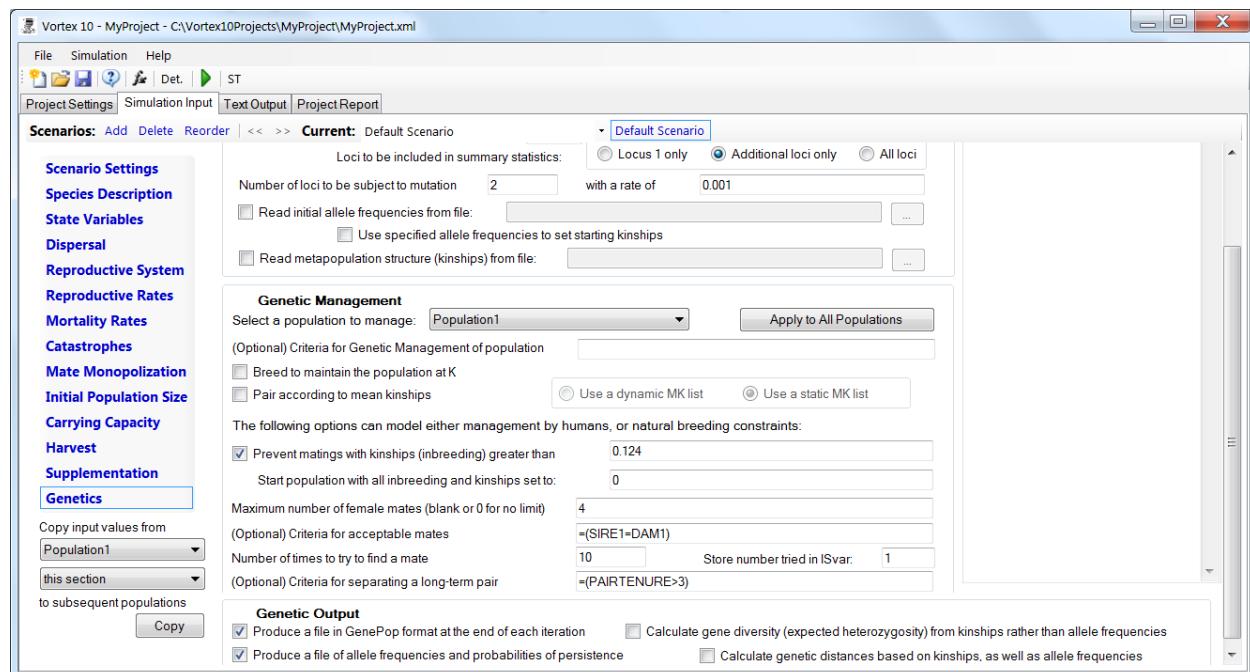
Note that for both Harvest and Supplementation, the number of individuals harvested or supplemented can be specified to be a function (e.g., of Year or N or some Population State variable) rather than a fixed number. If the function results in a non-integral number, VORTEX will use probabilistic rounding (see PROUND in the section on [Functions](#)) so that the mean number harvested or supplemented is not biased.

Genetics

VORTEX provides a variety of options to specify the initial genetic conditions, genetic management that might be applied (by either human managers or by the breeding behavior of the animals themselves), and more extensive genetic output. The default genetics model in *VORTEX* will simulate allele transmission at one neutral locus, and at a number of additional loci that are used to model the effects of lethal recessive alleles in inbred individuals. The maternally inherited mtDNA haplotype is also modeled.



Many users will have no need to specify any of the Genetics options below, as the default genetics model will meet their needs. As is the case with so many features in *VORTEX*, you can safely skip over this input section if you don't have any need for advanced Genetics tools.



Replace initial population with studbook from file: *VORTEX* can start the simulation with a known or assumed population that has a defined pedigree structure. This option allows the user to provide to *VORTEX* a population pedigree (a “studbook”) that might describe a captive population or a wild population that has been monitored closely for a number of generations.

For entering the initial population from a studbook, the studbook file can be any text file that contains a header of format:

ID;Sire;Dam;Sex;Selected;Alive;Age;Population;ISvars

followed by a line for each individual with data in the sequence given in the header. The .csv or .ped export file created by *SPARKS* (available from Species360, www.Species360.org) for *PMX* (available at <https://scti.tools/pmx>) can be used as the studbook input file in *VORTEX*, as can the

input file used for the Lineage pedigree program. The fields in the studbook are not required to be in the order shown in the above header, except that ID should always be the first field. *VORTEX* will use the header to determine the order of data fields in the studbook.

The output file with extension .all that is created when you choose the [Special Option](#) to produce a file of all individuals can also be used as the studbook input file for the starting population. Therefore, you can set this Special Option to produce the file from a single iteration, and then use the results of that simulation as the starting population for simulating further years. This is one way to start a simulation with a population that already has a pedigree structure and history.

The only required data in the studbook file are ID, dam, sex, alive, and age. If Dam or Sire is blank, WILD, or UNK for an individual, that individual is assumed to be a founder unrelated to all other founders. Sex should be coded as “Female” (or “F” or ‘f’ or “female”) and “Male” (or “male” or “M” or “m”). The Population field can be given either as a numeric value (1, 2, 3, etc.) that corresponds to the population sequence in the *VORTEX* project, or as the label specified for the population in *VORTEX*. If Population is missing, then all individuals are assumed to be in population 1. If either Selected or Alive is False (or 0), then the individual is assumed to be not in the living population at the start of the simulation (but it can still be used as an ancestor of other individuals in the genetic calculations). Age should be specified in years. Any age can be entered for dead individuals. ISvars should be at the end of the line of data, and the number of ISvars to be read from each input string is given by the “x” in undocumented option “Ix” (see [Special Options](#) above). Alternatively, if the header for the studbook file includes codes **IS1**, **IS2**, etc., or the labels for ISvars, then these ISvars will be read from the file. A value of -1 for an ISvar in the studbook file will be interpreted as a missing value code, to be replaced by the Initialization function for that ISvar.

Any lines in the studbook file that start with “*” or “#” are assumed to be comment lines and are ignored when individual records are read.

If you are modeling extra genetic loci, you can specify in the studbook data file what genotypes should be assigned to the individuals in starting population. This option is useful if you have a studbook population that has been genotyped at a number of loci and you wish to project how well those alleles are maintained into the future. This is somewhat like the option to specify starting allele frequencies (see below), but you further specify the exact genotypes of each initial individual. To do this, you specify in the header of the studbook file where the genotypic data will be, by using the codes **VV2**, **ZZ2**, **VV3**, **ZZ3**, etc. as the labels for the maternal and paternal alleles of extra locus 2, 3, etc., and **MT** as the label for the mtDNA haplotype. Note that these allele codes are the same as are used in functions that access alleles, and that locus 1 is always the default infinite-alleles neutral locus and you cannot specify the genotypes for that locus. Thus, to read in genotypes for 3 loci plus the mtDNA, your header might be:

ID;Sire;Dam;Sex;Selected;Alive;Age;Population;VV2;ZZ2;VV3;ZZ3;VV4;ZZ4;MT;ISvars

(You do not need to provide mtDNA haplotypes; just leave **MT** out of the header if you do not.) The ISvars, if any, need to be after the genotypic information in each line of data.

You also must specify that the number of neutral loci to be modeled (see below) is at least the number of loci with specified genotypes, plus one more for the default first locus. If you ask for more loci to be modeled than are included in the studbook file, the additional loci will be modeled with alleles assigned by *VORTEX*, just as normally happens. (If you ask for fewer loci to be modeled than for which genotypes are provided, the extra genotypes in the studbook file will be ignored.)

When coding the alleles in your studbook file, it is best if you use sequential numbers, starting with **1**, although you can use other positive integers if you wish. (Any numbers skipped in your coding sequence will be considered by *VORTEX* to be additional alleles that are not present in any individuals.) Entering an allele code **0** for a non-founder in the studbook will signify that the individual had not been genotyped and that *VORTEX* should assign alleles inherited from the parents at random. Founders should never be given the missing allele code of **0**, because *VORTEX* will have no way to assign alleles to those individuals. Any supplemented individuals in the population will be assigned new unique alleles, unless you specify the allele frequencies for the loci (see below).

Append studbook to initial population, rather than replacing initial N: By default, *VORTEX* will replace any individuals specified in the starting population(s) with those in a provided studbook. However, with this option you can add the studbook population to the default population specified in the [Initial Population Size](#) section. One way that this option can be used is to read one population (for example, a captive or otherwise intensively managed one) from a studbook, while letting other populations be started from default, unrelated individuals. If you use this option in this way, just be sure to set the initial N for the studbook population to 0, unless you do want that population to start with a combination of studbook individuals and new individuals.

Number of neutral loci to be modeled: *VORTEX* normally simulates Mendelian transmission of alleles at one neutral locus, at which every founder is assigned two unique alleles (i.e., it uses an “infinite alleles” model). Any number of additional loci can be simulated by specifying the number here. This will produce more precise estimates of genetic variation within and genetic distances among populations, and can be useful for simulating patterns of molecular genetic data that might arise from the population dynamics. Note that although the label says “neutral loci”, you can create fitness differences among genotypes by specifying that demographic rates are functions of the alleles carried by individuals.

There is not often any good reason to add more neutral loci to your simulation, unless you intend to specify the initial allele frequencies (see below) or model fitness differences between alleles. If you run enough iterations to obtain reliable demographic results, then the genetic results based on the default single locus will usually also be sufficiently precise.

Loci included in summary statistics: When you add loci to the genetic model, you can then specify if genetic statistics (such as heterozygosity) are to be calculated based on the default neutral locus, or only on your additional loci, or both. If you specify starting allele frequencies at your additional loci (see below), then you may want to have genetic statistics calculated only for those loci, or you may instead want the genetic statistics calculated only for the default infinite alleles locus.

Note that the simulated loci have no effect on the calculations of inbreeding of individuals, as individual inbreeding coefficients are calculated from the pedigree, and the impacts of inbreeding are determined by the calculated inbreeding coefficients (for the portion of inbreeding depression not caused by recessive lethal alleles) and by additional loci (internal to the model, other than those discussed above) that are used to model the effects of recessive lethal alleles. The “inbreeding” of the population (for example, if you use ‘I’ as a variable in some population state variable) is calculated as the homozygosity at the modeled neutral loci, so it is influenced by which loci you ask to have included in summary statistics.

Mutation: Mutation rates can be specified for modeled loci. (The default is no mutation.) The mutation rate (per individual per allele) is specified. The user can also specify which loci are subjected to mutation. If no allele frequency file has been provided (see below), an infinite allele model of mutation is used. If an allele frequency file is provided, then a finite allele mutation model is used, with the mutation being sampled at random from the other alleles at the locus, as specified for the “supplementation”, or last, population.

By using functions for the mutation rate, you can specify different rates for different loci, or different rates for the maternally vs paternally derived alleles. In the function for mutation rates, LOCUS indicates the affected locus and the variable MALLELE is 1 for the maternal allele and 0 for the paternal allele. Thus, a mutation rate of $=(\text{LOCUS}=1)*0.0001+(\text{LOCUS}=2)*0.0005$ will apply a per generation mutation rate of $m = 0.0001$ for locus 1 and $m = 0.0005$ for the 2nd locus.

Mutation of the mtDNA can also be specified, by using LOCUS= -1 in any mutation rate function and asking for mutation to be modeled at nLoci+1 loci (one more than the number of neutral loci being modeled).

Read initial allele frequencies from file: VORTEX normally starts the simulation with each founder individual carrying unique alleles at the modeled locus or loci. For testing changes in known or assumed starting frequencies of alleles at loci, you can specify that the simulation should sample from a known distribution of allele frequencies when it assign alleles at your additional loci to founders. This option might be used to test expected patterns of microsatellite DNA alleles, or to test the evolution of alleles with specific selected effects on demography. (Although the alleles at “neutral” loci in VORTEX usually have no impact on fitness, it is possible to specify that demographic rates are functions of genotypes.) The starting allele frequencies are specified in a text file with the following format:

```
3
3 4 2
.33 .33 .34
.25 .25 .25 .25
.5 .5
```

The first line specifies the number of loci modeled. The next line gives the number of alleles for each of these loci. The subsequent lines give the allele frequency distributions for the loci.



Note that, as for all input values in *VORTEX*, the decimal delimiter in these files must be entered as a comma, rather than a point, when the computer is using European or many other non-American data formats.

If you specify allele frequencies for more loci than were in your model (see above), then the last locus in the allele frequency file will be used to set the initial haplotypes for the mtDNA. If you specify alleles for more loci than needed for your requested number of neutral loci plus the mtDNA, then additional neutral loci will be modeled, with the frequencies you have specified. (The starting frequencies for mtDNA alleles will still be taken from the last locus in the file.)

If you are modeling more than one population, you can repeat the rows of data for each population. Here is an example of a file with allele frequencies at three loci for three populations:

```
3
3 4 2
.33 .33 .34
.25 .25 .25 .25
.5 .5
.4 .3 .3
.5 .5
.25 .75
.3 .3 .4
.35 .15 .35 .15
.75 .25
```

If you have more populations in your model than data in the allele frequency file, then *VORTEX* will use the last set of frequencies for all subsequent populations. Thus, if you want all populations to start at the same frequencies, you can just give that set of frequencies once in the file. When you have supplements, *VORTEX* will use the frequencies specified for population $nP + 1$, in which nP is the number of populations. Again, however, you can omit specification of an extra set of frequencies for any supplements, and then *VORTEX* will use the last set of frequencies that you did provide. Note that if there are not enough frequencies specified for the numbers of alleles specified on line 2, then *VORTEX* will fill in the remaining alleles with frequencies of 0. that you did provide. Note that if there are not enough frequencies specified for the numbers of alleles specified on line 2, then *VORTEX* will fill in the remaining alleles with frequencies of 0.

VORTEX will re-scale allele frequencies at each locus so that they add to 1.0. This allows you to provide proportional frequencies (or counts) and not worry if they do not sum to 1.0.

How can I project how well rare (and common) alleles will be retained in my population?

Because VORTEX starts each founder with two unique alleles, the number of alleles persisting over time can be difficult to interpret. Most alleles in real populations don't start out being present in just one copy (except right at the point that they arise by mutation), and such singleton alleles are lost with very high probability even in large populations. Therefore, the number of such alleles reported by VORTEX makes it look like diversity is being lost fast (and it is, for any initially singleton alleles), even while the population is rather healthy genetically overall.

To determine the loss of alleles that start at higher frequencies, you could add together a number of founder alleles (in the optional genetic output file that reports allele frequencies and persistence, see below) to get the frequencies and probabilities of loss of aggregate sets of founder alleles. But this is a really hard way to estimate loss of alleles: you would need to determine how many founder alleles to combine; the probability of cumulative loss is a complicated function of the losses of individual alleles; and the calculations are a lot of work.

The much easier method to determine probabilities of allele loss for alleles starting at various frequencies is to add one or more additional loci, assign your desired allele frequencies to them, and then track how they do. For example, if you want to know the probability of loss of rare alleles of initial frequency 0.05, use the methods above to assign **0.05** as the frequency to 20 alleles of a locus.

You can simultaneously test the rate of loss of alleles of various frequencies by assigning the frequencies to be tested to your additional loci. For example, giving initial allele frequencies of **0.05 0.05 0.10 0.10 0.20 0.50** will test the rate of loss of initially rare, mid-frequency, and common alleles.

Read metapopulation structure (kinships) from file:

This option was added to facilitate specifying the initial genetic structure of a metapopulation. A file can be read that specifies the kinship of individuals in each population to those in other (or the same) population. When the file is read, VORTEX will create for you functions in the “*Start population with all inbreeding and kinships set to:*” box (see below) for each population. For example, if the file is:

```
# test population kinship matrix  
0.1; 0.05; 0.01  
0.05; 0.2; 0.06  
0.01; 0.06; 0.15
```

then the functions specifying the initial kinships will be (for the three populations):

$$\begin{aligned} &= (\text{PY}=1)*(0.1) + (\text{PY}=2)*(0.05) + (\text{PY}=3)*(0.01) \\ &= (\text{PY}=1)*(0.05) + (\text{PY}=2)*(0.2) + (\text{PY}=3)*(0.06) \end{aligned}$$

and

$$=(PY=1)*(0.01)+(PY=2)*(0.06)+(PY=3)*(0.15)$$

The values given in the file should be separated by semi-colons, spaces, or tabs (but *not* commas). Logically, the matrix should be symmetrical, but *VORTEX* will not force it to be.

Genetic Management options

Select a population to manage: Genetic management options can be applied to all populations or to any subset of populations. Different options can be specified for each population. To specify a set of Genetic Management options for a population, select that population from the dropdown list. Be sure to select your population *before* changing the Genetic Management settings for that population! After you set the Genetic Management options for a population, you can copy those settings to all the other populations by hitting the “Apply to All Populations” button. Even after copying settings to all populations, you can select a specific population from the dropdown list and change one or more of its options to be something different from the options that were copied to all populations. In this way, if you are careful, you can relatively quickly set the options that are to be the same for all populations (do this first!) and then set some specific alternative options for each population. After you set Genetic Management options for several populations, it is a good idea to click through the dropdown list of populations to confirm that the options are correctly set to what you want for each one.

Optional criteria for Genetic Management of Population: You can provide a function that will specify under what conditions the genetic management options will be applied. When the function evaluates to True (1), the options are applied; when it is False (0), they are not. For example, you can set a function of **=N>(0.75*K)** to have the Genetic Management applied only when the population is above 75% of K. These criteria will not affect the option to set initial kinships to a value other than 0.

Breed to maintain the population at K: With this option, the program will calculate each year the expected number of matings required to bring the population to (but not beyond) the carrying capacity. Note that K may still not be reached if there are not enough adult females to fulfill the required number of matings. When this option is selected, the “% adult females breeding” (in [Reproductive Rates](#)) has a somewhat different meaning than usual: it will be the maximum % of females that can breed, but the option to breed to K may result in fewer females breeding. The calculation of the numbers of matings to maintain the population at K takes into consideration the percent of breeding females that will produce each possible number of broods, the mean brood size, and the survival rates. It does not adjust for possible inbreeding depression.

Pair according to mean kinships: Managed breeding programs can retain maximal genetic diversity by selecting as breeders those individuals with the lowest mean kinship (MK) to the living population (Ballou and Lacy 1995). If this option is selected, then the user must also specify if the MK list for prioritizing breeders is to be updated as each pairing is selected (“dynamic MK list”) or left unchanged within each year as pairs are selected (“static MK list”). A dynamic list will preserve genetic diversity better if most males and females that are paired do produce offspring (i.e., the % females breeding is high). A static list may provide better genetic

management if many pairings fail (because the dynamic list would incorrectly assume that many pairs do produce the expected offspring). Note that selecting breeders to minimize MK will not usually be effective unless the “Breed to maintain at K” is also selected. This is because all females (and possibly most or all males) will be used for breeding, so selecting the lower MK individuals first for breeding still will not give them more mating opportunities. There is a [Special Option](#) that can be used to prioritize breeding on traits other than MK.

Prevent matings with kinships (inbreeding) greater than F = x.xxx: With this option, females will not be paired with a male that is more closely related than the specified maximal allowable inbreeding. If a male initially chosen is too closely related, a different male will be picked instead. (See the option four down to see what happens if repeated males are unsuitable.)

Start population with all inbreeding and kinships set to: This option allows initial inbreeding coefficients and pairwise kinships among founders to be set to a value greater than 0. With this option, you can start with a population that is already inbred and low in gene diversity. This will cause inbreeding depression to be imposed immediately on their offspring (if inbreeding depression is chosen, and if not all of the inbreeding depression is due to recessive lethal alleles). Note that the higher inbreeding that occurs when this option is chosen will not affect the model of inbreeding depression due to recessive lethal alleles, because that part of the inbreeding depression is not calculated based on inbreeding coefficients, but rather is a consequence of the simulated transmission of recessive alleles. Note also that this option will also not affect the mean expected and observed heterozygosity, or any calculations on the population mean inbreeding (which is always set at the mean homozygosity of modeled loci), but it will affect the inbreeding of individuals. The mean inbreeding of individuals can be determined by setting IS1 =**I**, and then setting PS1 =**IMEAN1**.

The optional value to which initial kinships are set can be a function that includes a distribution (by using a **RAND** function), the populations of the two individuals (**PX** [or just **P**] = population of first individual; **PY** = population of second individual), the ISvars (**X1** [or **IS1**], **X2**, **X3**, etc. = state variables for the first individual; **Y1**, etc. = state variables for the second individual), or anything else (e.g., **Y** or **R**). This allows you to set initial kinships to be variables from a distribution, to be different for within vs between population pairs, to change during a simulation (e.g., higher for new supplements that arrive later), or to be functions of some property (e.g., age) of the individuals.

Note that every pairwise kinship among founders is normally set to the same value as the inbreeding coefficients of founders. This should be approximately correct for randomly breeding populations because the inbreeding coefficient is equal to the kinship between the parents. However, in small populations the inbreeding coefficients will, on average, be a little less than the average kinship, because the average kinship increases slightly from generation to generation. Also, it may be that individuals actively avoid inbreeding, in which case inbreeding might be much less than the average kinship.

If you want to set the initial inbreeding coefficients to be different than the initial pairwise kinships, there is a trick that you can use. Set the initial kinships to

=(*I*=0)*inbr+(*I*>0)*kin

In which inbr is the value to which you want to set the initial inbreeding coefficients, and kin is the value of the initial kinships. This trick works because inbreeding starts as the default 0 and then it is calculated (so it will be set to inbr when an individual is first created), and then kinships to all other individuals are determined. (Note that if you want to set the initial inbreeding to 0 but the initial kinships to something else, then you need instead to set inbr to some very tiny number (such as 0.00001), so that *I* > 0 will trigger the setting of kinships to kin.)

There are three ways in which *VORTEX* can determine the kinships among individuals and thus the inbreeding coefficients of their offspring, and it is important to know how these methods interact (and which takes precedence). The default assumption is that founders are unrelated to each other and non-inbred, and then the kinships among later generation individuals are determined from standard pedigree analysis. However, as described above, the initial kinships and inbreeding coefficients can be specified by the user. If this option is chosen, then these specified kinships replace the values of 0 that are first assigned to founders. The third method is that within [Special Options](#), you can specify that the initial kinships of a studbook population (or a subset of it) are to be read from a file. If such a file is provided, these kinships will be used in place of any default kinships or the values given in the above Genetic Management option. All three of these methods can be involved in the setting of the initial kinships of a Scenario if, for example, some kinships are specified for studbook individuals, other kinships for founders of one or more populations are specified in the Genetic Management option, and the default initial kinships of 0 are left in place for yet other populations in the Scenario.



Keep in mind that kinship coefficients and inbreeding coefficients calculated from them will impact inbreeding depression only to the extent that inbreeding depression is due to heterozygote advantage rather than recessive lethals. (See [section on Inbreeding Depression](#).) The recessive lethal alleles are modeled with simulated loci, with each founder carrying unique lethal alleles, and frequencies of lethal alleles are not affected by what you specify for initial kinships and inbreeding.

Maximum number of female mates: The option allows you to put a constraint on the number of females to which a male can be paired each year. This option can be useful to constrain harem size, or to prevent a single male from dominating breeding when pairs are selected by using a static MK list.

(Optional) Criteria for acceptable mates: This option allows the user to specify a constraint on which males are suitable as sires for each female. The constraint is specified using a function. For these functions specifying mate suitability, the standard variables describing individual characteristics (e.g., 'A' for age, 'I' for inbreeding, or PARITY) refer to the values for the potential sire. You should use SIRE1, SIRE2, etc., and DAM1, DAM2, etc. to indicate the IS variables for the sire and dam. If you want the function to include a characteristic (such as Age) of the dam, then you need to set and ISvar to that variable (e.g., IS1=A) and refer to DAM1 in the function. If the function evaluates to a number between 0 and 1, then the result will be used as the probability that the male is accepted as the mate.

Examples of the use of this option include:

=I<0.10 [for avoiding mating with a sire than has an inbreeding level greater than 10%]
=SIRE1>DAM1 or =A>DAM1 [to force the sire to be older than the dam, if IS1 is set to age]

This option can provide a great deal of flexibility in creating complex breeding systems, but it can be difficult to specify the constraint function correctly.



Be careful how you specify the potential mates and their variables, because it can be confusing (and was sometimes handled differently in earlier versions of VORTEX !) which individual's traits are being referenced. **DAM** (or **DAMID**) and **SIRE** (or **SIREID**) generally refer to the female and male being tested for acceptability as a pair – not the dam and sire of either of these potential mates. An exception to this occurs if you use **DAM** as an Individual State variable (for changing the “dam” of an individual!), in which case you need to use **DAMID** to indicate the female being tested as the dam (because the ISv **DAM** now refers to the dam of the male being tested).

Number of times to try to find a mate: If constraints on suitable males to select as mates are specified in the above options related to inbreeding or limitations of pairings, then this option will specify how many times the program should select a male and test its suitability before giving up and leaving the female without a mate for that year. A default value of 10 is used if this input box is left empty or is set to **0**.

Store number tried in ISvar: If constraints on suitable mates are specified in the above options related to inbreeding or limitations of pairings, then you can store the number of times that each female tried to find a mate (in the most recent breeding cycle) in an ISvar. This value will be one more than the maximum number of tries allowed (see above) if no mate could be found for that female. This option has no effect on this ISvar for males or non-breeding females.

Optional criteria for separating a long-term pair: Although Long-term Monogamy and Long-term Polygyny (specified on the Reproductive System input page) normally leave a pair together until one of the mates either dies or disperses to another population, you can specify that pairs remain together only for set tenure. A function or number entered for this option will be used as the probability each year that a long-term pair will be separated. For example, if a function evaluates to 0.25, then with 25% probability each year the pair will be separated and new mates assigned.

Genetic Output

Four options exist for producing more detailed or different genetic output for analysis.

Produce a file in GenePop format at the end of each iteration: Checking this box will generate at the end of each iteration of the simulation a file (with extension .gp) listing all living individuals and their genotypes (at those loci for which summary statistics are requested). The file is in the format required for analysis with the *GENEPOP* software (<http://wbiomed.curtin.edu.au/genepop/>).

Produce a file of allele frequencies and probabilities of persistence: If this option is checked, then *VORTEX* will create an output file (with extension .gen) that lists for each founder allele the mean frequency at the end of the simulations and the probability of persistence (proportion of iterations in which the allele persisted).

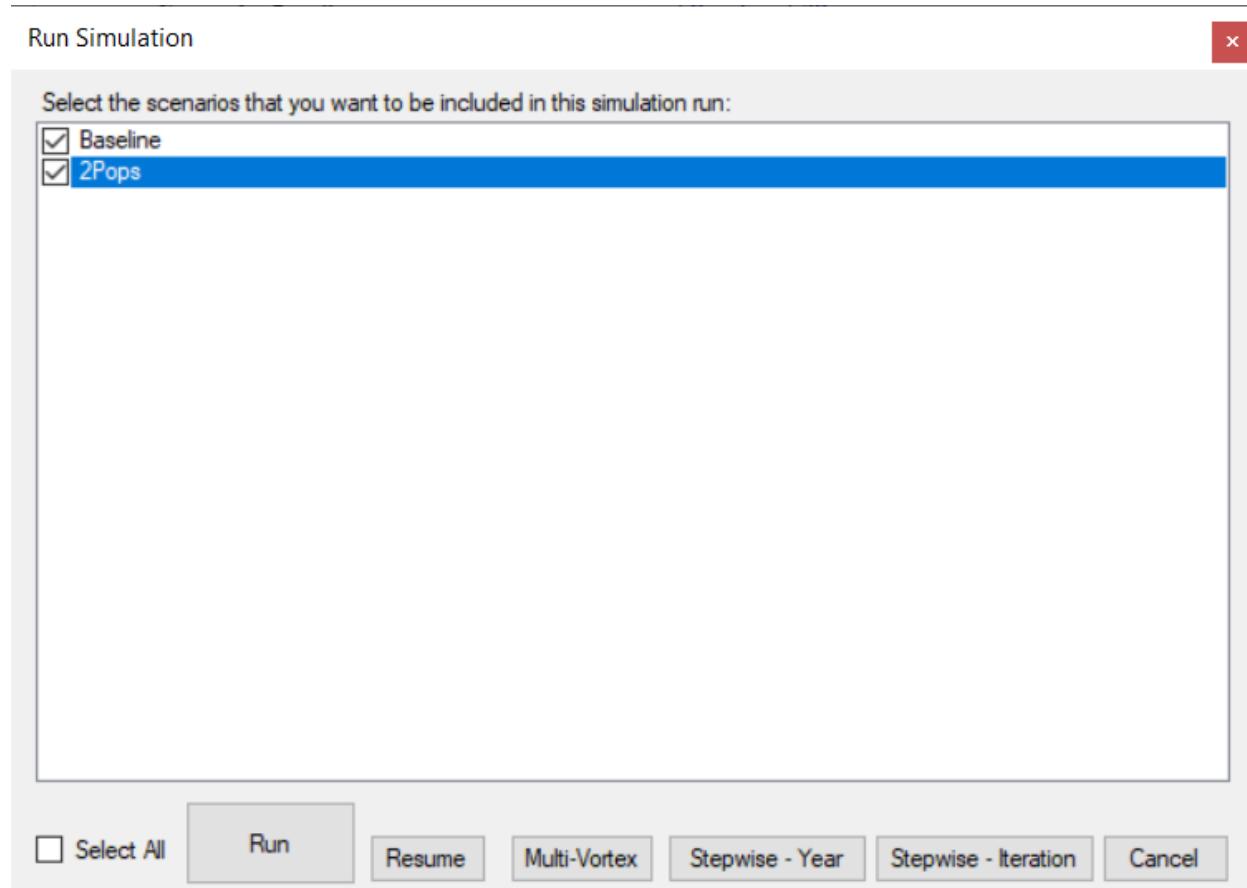
Note that alleles are listed in the order in which they were assigned to initial animals. Thus, if your model had 3 populations, each started from 10 individuals, then the first 20 alleles descended from founders of population 1, the next 20 from population 2 founders, and alleles 41 to 60 from population 3 founders. Also, any supplements added during the simulation would receive alleles starting with the 61st one in the list. This allows you, for example, to determine how much genetic mixing occurs between populations during a simulation, or to determine what percent of the gene pool descends from supplemented individuals. Although you can analyze the optional Genetic Output files to obtain this information, often it is easier to set Individual State variables to determine the source of each individual's alleles and then use Population State variables to tally these contributions from various founder sources. *VORTEX* automatically tallies the mean, SD, and SE for each Population State variable in the model.

Calculate gene diversity (expected heterozygosity) from kinships rather than allele frequencies: This option causes the calculations of gene diversity (expected heterozygosity) to be based on the kinships, rather than on the allele frequencies at simulated loci (as is otherwise done by *VORTEX*). When initial kinships are 0 (the default), these two methods of calculating gene diversity will yield very similar results. However, if you specify different kinship structure among the initial individuals (via the Genetics option to specify all starting kinships, or via the Special Option KM), then it will often be more useful to see the gene diversities that arise from those initial kinships. Note that the initial kinships have no effect on the initial alleles assigned at the modeled loci, and thus have no effect on genetic metrics calculated from allele frequencies.

Calculate genetic distances based on kinships as well as allele frequencies: For scenarios with more than 1 population, this option will include in the Output Summary genetic distance matrices based on the kinships, as well as the standard genetic distance matrices based on allele frequencies.

Running your simulation

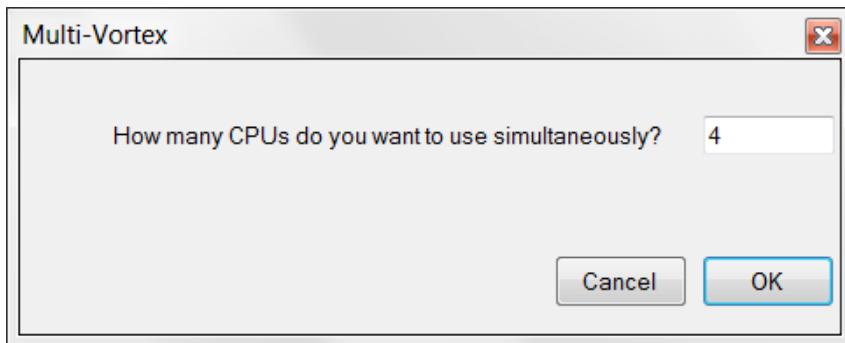
Now that you have entered all the data for your scenario, you can run your simulation! Click on the Run icon (the green triangle), or select Run from the Simulation menu, or hit F5 to open the Run Simulation window.



Check the Scenarios that you want to run, and then hit the “Run” button. During the simulation, *VORTEX* will display a graph of the changing size (unless you had selected the Special Option to not show graphs during iterations). *VORTEX* erases the lines after each 100 iterations and starts again, because drawing the lines can be the slowest part of the simulation when there are many lines on the graph, and because the graph usually becomes unreadable with more than 100 lines.

The “Resume” button allows you to add iterations to a scenario. For example, you can run 500 iterations, then increase the number of iterations to 1000, and Resume the simulation to add iterations 501 to 1000 to the outputs of that scenario. Note that some of the more detailed genetic output, including the allele frequencies, probabilities of allele persistence, and measures of genetic distance between pairs of populations will not be averaged over both the prior and the new (resumed) iterations, but instead will be reported only for the new iterations. All other output will show the results averaged across both the prior and the new iterations.

The “Multi-Vortex” button allows you to run multiple scenarios simultaneously by using different CPUs on your computer. This can allow batches of scenarios to be run more quickly than when they are run sequentially. Because of the extra over-head of work needed for *VORTEX* to coordinate this multi-tasking, running on 4 CPUs, for example, will not be 4x as fast as running the scenarios in the normal Run mode, but it will likely be 2x or 3x faster. When you select this option (which will have no effect if you are running just a single scenario), *VORTEX* first asks if it is OK for it to save your project file. It needs to save the project first, because the multiple scenarios will be run via a command-line version of *VORTEX* that runs outside of the user interface and accesses the project file. It is possible that you were just testing some changes to the project and do not want to save those changes, in which case you should not run in Multi-Vortex mode. (In that case, just “Cancel”.) Assuming that you are OK to have the project file saved, *VORTEX* then asks “How many CPUs do you want … ?” The default value will be half the number that *VORTEX* detects on your system, as that is usually an efficient number to use. (It leaves some processor power available for other tasks in Windows.) But you can select any number that you want, even more than the number of CPUs on your system. If you do select more than is on the system, then Windows will need to have several instances of *VORTEX* sharing CPUs, and usually this results in no more speed (and sometimes less) than if you don’t force multiple *VORTEX* to share CPUs.



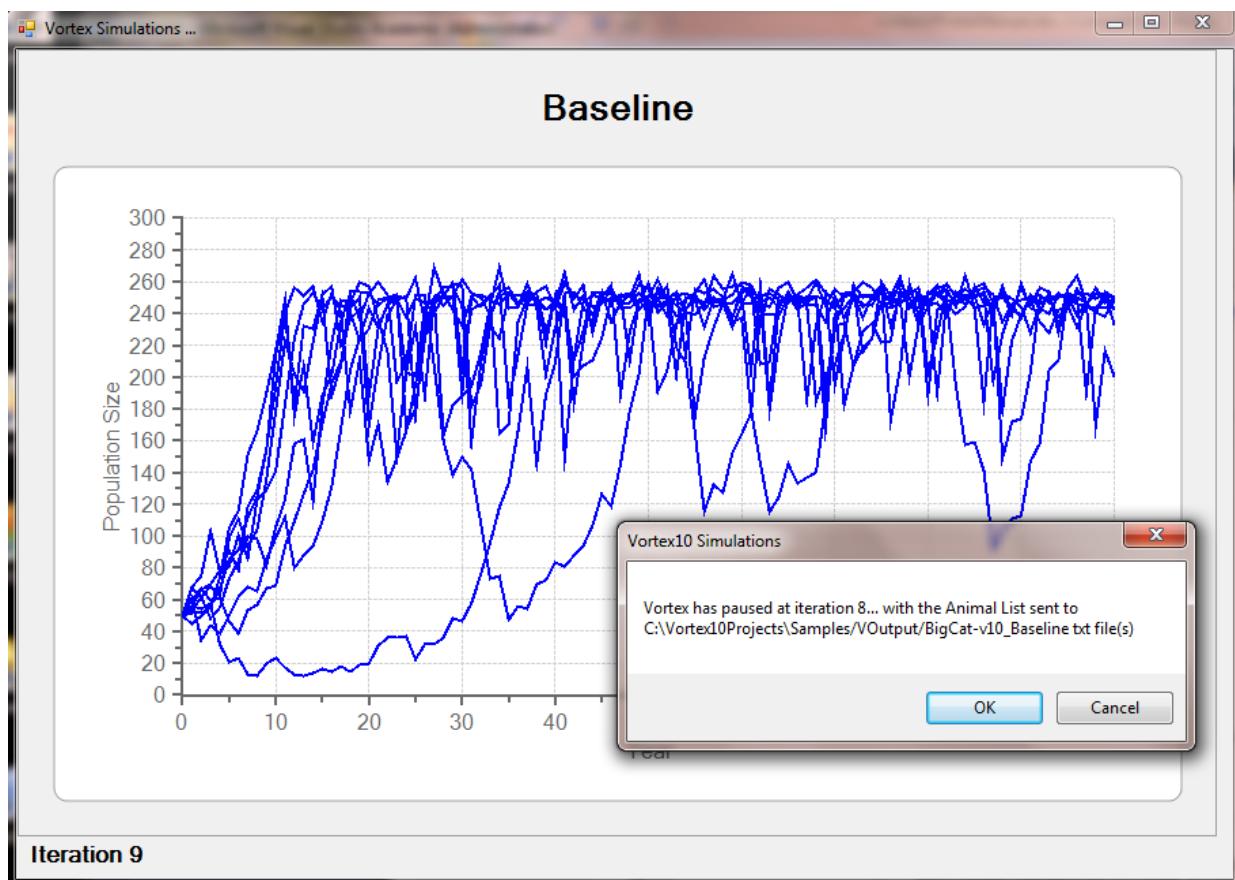
Note that if *VORTEX* is running on a virtual Windows machine on a server (or networked set of computers that can share tasks), it is possible that it can access many CPUs for the work. (In this case, *VORTEX* might not be able to detect the number of CPUs available, so the default might not be correct.) For example, a large number of *VORTEX* scenarios can be run simultaneously on Microsoft Azure cloud-based Windows servers. Someday soon, an appendix will be added to this manual to explain in more detail how to configure *VORTEX* to run on such platforms.

When running in Multi-Vortex mode, it usually is sensible to select the Special Option “Do not show graphs during iterations” on the Project Settings tab. This is because the graphs for the simultaneously running scenarios will be on top of each other (although it is somewhat fun to see them all working away); showing graphs slows down the simulation (while the only reason for running in Multi-Vortex mode is to gain speed); and on some computers the graphics card might not be able to keep up with the multiple scenarios that are running (so you might not see the graphs anyways). Unless you also select the Special Option “Do not show messages during run”, *VORTEX* will show a message to let you know when all the scenarios have finished.



Do not start several instances of Vortex user interface and then simultaneously run multiple scenarios of the same project using the Multi-Vortex option in each. This will cause file conflicts between the several Multi-Vortex batches of runs. You can run multiple instances of Vortex simultaneously if you are not using the Multi-Vortex mode for more than just one of them, or you can run several instances of Vortex, each running in Multi-Vortex mode, if they are working on different projects.

The “Stepwise” buttons allow you to run the simulation either one year at a time or one iteration at a time, with the program pausing in between until you tell it to proceed by hitting “OK”. At each paused step, *VORTEX* writes a list of all individuals in each population to a text file. These stepwise runs can be useful for teaching, or for confirming that at each step of the simulation, things are proceeding as you expect. If you hit “Cancel” when the program has paused, then *VORTEX* will no longer pause the simulations and will proceed with all the iterations.



After the simulation has completed, *VORTEX* will show a few summary statistics at the bottom of the graph. You can Save or Print the graph, or edit it further by accessing graph Properties by double-clicking on the graph. Close the graph when you are ready to return to the main interface.

If you want to stop *VORTEX* in the middle of a simulation (for example, if it is running very slowly and you think that you had entered something incorrectly and will need to run it again), you can try to click (often multiple times) on the ‘x’ icon in the corner of the graph of N being generated during the simulation. Often this will work, because it forces (sometimes after a little

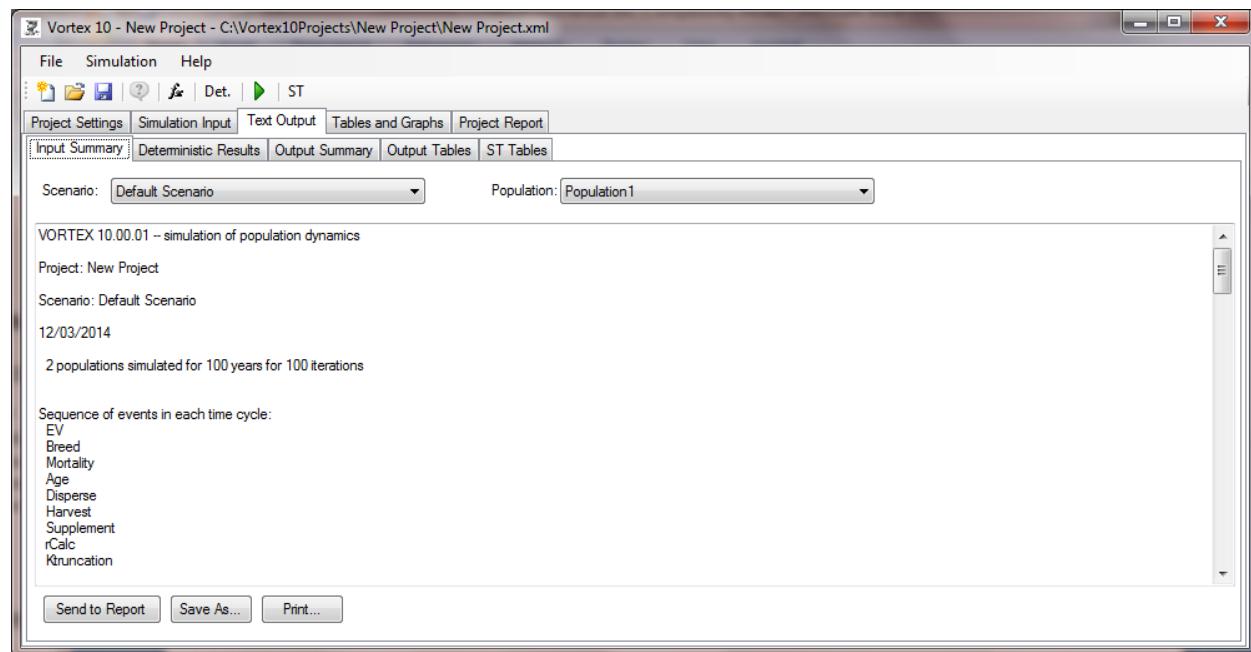
time) *VORTEX* to crash when Windows can't continue to update the graph. If this does not work, and you are still desperate to stop *VORTEX*, you can open the Windows Task Manager (accessible with **Ctrl-Alt- Del**), click on the *VORTEX* Application, and then End Task. Either of these methods is rather drastic, and Windows may give you a meaningless message about trying to find a solution or sending a report about the crash. (Microsoft has no solution to provide, and it won't do anything with such reports.) *VORTEX* will shut down, and it may leave partial and corrupted output files in your project folder. This is a good reason to always save your Project after you make changes, before you run the simulation! If you do force *VORTEX* to crash, or it does so on its own, it is best to fix the data and rerun the simulation (even with just a few iterations) so that it will create valid output files. Otherwise, when you go to look at tables and graphs, *VORTEX* might crash because of trying to access corrupted files.

Text Output

The Text Output provides five tabs for viewing text and tables with data. The text and tables that can be viewed in the Text Output tab are all saved as text files in the VOutput subfolder of the Project folder. There are other output files (both standard and optional) that are also placed into the VOutput folder for possible later viewing and analysis, but the user interface does not display them. For each of the tabbed sections of text displays, you can send the text to the Project Report, save a copy of it under a new file name, or print the text file.

Input Summary

The first tab shows a text file (saved with extension .inp in the project folder) that has all of the input values for each scenario. You can move between scenarios with the Scenario dropdown list. Within a scenario, you can jump to the beginning of the text for a population by selecting it from the Population dropdown list. You can also scroll to any place within the input text.

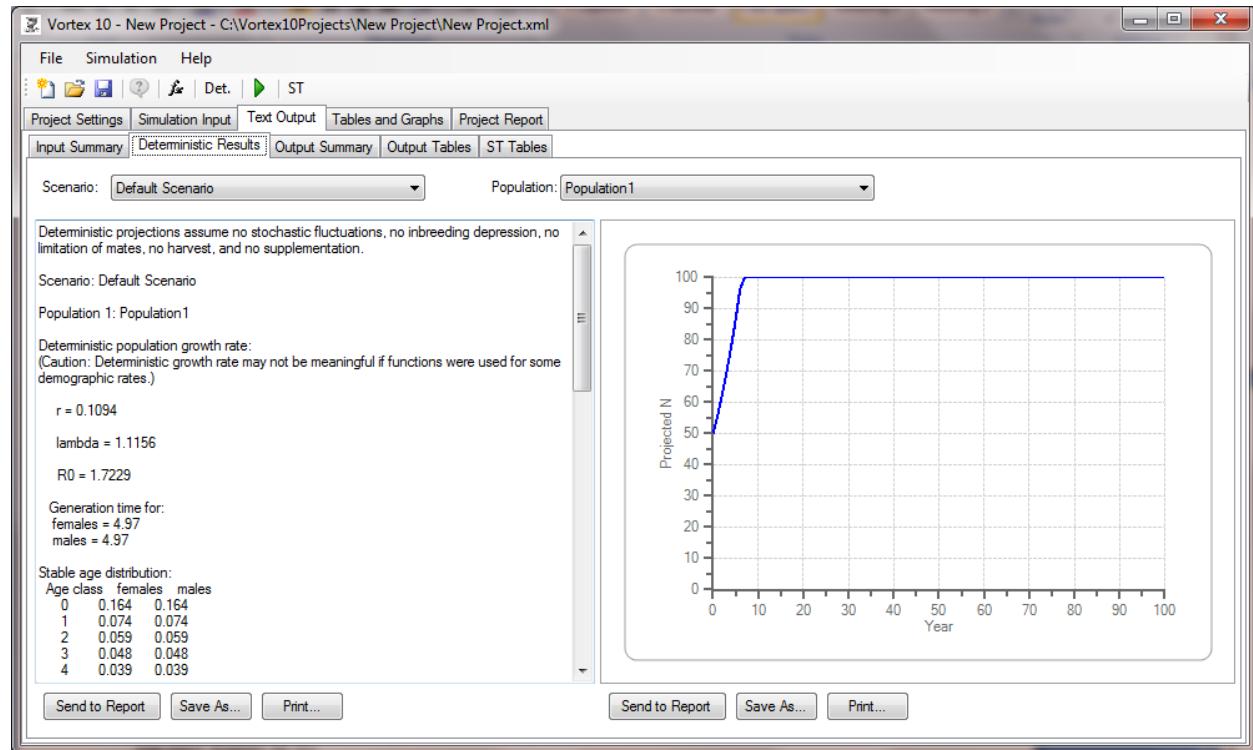


The Input Summary will not be updated until you save your Scenario.

It is always a good idea to scroll through the Input Summary to be sure that you entered all the input values correctly for your scenario, and that *VORTEX* interpreted the inputs as you intended.

Deterministic Results

The second section of Text Output provides both text and a simple graph to display the deterministic projections of population size. The text window shows the exponential rate of increase, r , the annual rate of change, λ , and the per generation rate of change or “net replacement rate”, R_0 , as determined from life table analysis of the mean rates of reproduction and survival in your model. The mean generation time and a stable age distribution (calculated from age-specific birth and death rates) are also given.



The graph given with the Deterministic Results is fairly simple and crude, but it shows the exponential growth (or decline) projected from the life table calculations (up to the limit set by the carrying capacity). The graph can be sent to your Project Report, printed, or exported to a file for import into other programs. As with all graphs in *VORTEX*, double-clicking on the graph opens Chart Properties in which you can customize the graph.

The deterministic calculations are performed by *VORTEX* as soon as you save the input values for your model. Therefore, you can view these results (and also the Input Summary text) even before you run your simulations. It is often helpful (and always recommended) to look at the deterministic projections before proceeding with stochastic simulations, so that you will know whether the rates of reproduction and survival are at least minimally adequate to allow for population growth in the absence of random fluctuations and other destabilizing processes (such as inbreeding and harvest). Remember that you can also see a shorter version of the Deterministic Results while you are entering input values by hitting **ctrl-D**.



It is important to look at the deterministic projections of population growth for any analysis. If r is negative, the population is in deterministic decline (the number of deaths outpace the number of births), and will become extinct even in the absence of any stochastic fluctuations. The difference between the deterministic population growth rate and the growth rate resulting from the simulation can give an indication of the importance of stochastic factors as threats to population persistence.

Deterministic Calculations in VORTEX

Before the actual stochastic simulation begins, VORTEX performs a standard life table analysis to calculate the deterministic mean population growth rate (r , the exponential growth rate; or λ ("lambda"), the annual multiplicative growth rate), the mean generation time for males and females, and the stable age distribution used (optionally) to initialize the starting population. These calculations will provide accurate long-term averages, if stochastic variation (due to demographic stochasticity, environmental variation, catastrophes, and inbreeding effects) is minimal. Life table analyses implicitly assume that age-specific birth and death rates are constant through time; they yield over-estimates of long-term population growth if there is any variation in demographic rates.

The deterministic population growth rate r is calculated by solving the Euler equation:

$$\sum(l_x m_x e^{-rx}) = 1,$$

in which l_x and m_x are the age-specific mortality and fecundity rates, respectively for age class x to $x+1$, and the summation is over all age classes. Lambda is related to r by:

$$\lambda = e^r.$$

The stable age distribution, or the proportion of the population at each age class c_x , is given by:

$$c_x = \frac{l_x e^{-rx}}{\sum(l_x e^{-rx})}.$$

The determination of a stable age distribution for males in a given population becomes a bit more complicated if the mortality schedules are different for the two sexes. In this case, r is calculated based on female life history tables (since females usually control population growth), but the l_x 's used in the age distribution equation are those for males. Moreover, the exact form of the equation is dependent on when the age classes were censused. In the above equation, c_0 is the proportion of the population aged 0 plus a small increment, just after the breeding season but before mortality is imposed. In order to build the initial population, VORTEX omits age class 0 (because the simulations begin just before the breeding season), and re-scales the age distribution in order to sum to 1.0.

The life table calculations assume that there is no limitation of mates (i.e., there are always enough breeding males to mate with all breeding females). Other complications arise if there are catastrophes in the simulation model. In those cases, VORTEX adjusts the fecundity and mortality rates to account for the effect of catastrophes, averaged across years.

For more information on the details of life table analysis, refer to any number of general ecology or population biology texts such as Pielou (1977), Krebs (1994), or Caughley (1977).

Why does the population growth rate in the simulation (stochastic lambda) almost always end up smaller than the growth rate calculated from the birth and death rates (deterministic lambda)?

There are several reasons for this, and they are some of the reasons that stochastic simulations are valuable!

First, the lambda calculated from vital rates is an expected mean growth rate if all vital rates are constant at the values entered into the model. However, if birth and death rates fluctuate over time (even due just to random chance), as they always do in the field and as they will in any stochastic simulation model, then the lambda that is calculated from the means will systematically overestimate long-term average population growth. The reasons for this are complicated, but basically arise from the fact that good and bad years don't cancel each other out. E.g., if a population alternates between $\lambda = 1.1$ and $\lambda = 0.9$ in alternating years, the long term growth rate will be < 1.0 . (The same thing happens with survival rates: alternating between 80% and 90% survival is not as good as a constant survival of 85%, because survival compounds in a multiplicative way, rather than in an additive way.) The population ecology textbooks usually don't tell the students that the standard life table methods are systematically biased toward overestimating population growth! (This oversight occurs because the methodologies came from human demography, in which the population sizes are so large that there is almost no fluctuation in rates over time, so the bias is tiny in human population projections.)

Second, lambda projects the population growth *if* the population starts at a stable age distribution. If instead the population currently has an age distribution that will result in a temporary baby bust (a lack of prime breeding age animals), then the population can decline for a number of years until that age distribution settles down to the expected long-term average.

Third, calculations of deterministic lambda ignore a number of density dependent processes that can depress growth at small population sizes. These include inbreeding effects and also things as simple as random fluctuations in sex ratio sometimes making it hard for all the females to obtain mates.

Output Summary

The third section of Text Output lists the basic status of each population at each year of the simulations.

The statistics reported in this file (saved with extension .out), for each Scenario and each Population, are:

- The cumulative number of iterations in which the population is extinct or remains extant;
- The probability of population extinction (PE) or survival (equivalent to the proportion of iterations that the population is extinct or remains extant);
- The mean population size, reported separately for all populations (N-all) and only for those remaining extant (N-extant), with standard error (SE) and standard deviation (SD) across iterations;
- The mean “gene diversity” (or “expected heterozygosity”) remaining in the extant populations, with standard error and standard deviation across iterations;
- The mean “observed heterozygosity” (equal to $1 - [\text{mean inbreeding coefficient}]$) remaining in the extant populations, with standard error and standard deviation across iterations;
- The mean number of alleles remaining within extant populations (from an original number equal to twice the number of founder individuals), with standard error and standard deviation;
- The mean number of mitochondrial haplotypes remaining within extant populations (from an original number equal to the number of founder individuals), with standard error and standard deviation;
- If the inbreeding depression option is included in the simulation, the number of lethal alleles remaining per diploid individual, with standard error and standard deviation, determined by the nature and extent of the genetic load identified in the input process, and the intensity of inbreeding the population undergoes.

Note that a [Special Option](#) allows the genetic statistics above to be calculated across all runs – extant and extinct.

Following these yearly reports, the output file presents a series of final summary information that includes:

- The final probability of population extinction and, the converse, the probability of population persistence;
- If at least 50% of the iterations went extinct, the median time to extinction;
- Of those iterations that suffer extinctions, the mean time to first population extinction, with SE and SD across iterations;
- The mean times to re-colonization and re-extinction of those simulations that went extinct;
- The mean final population size, with SE and SD across iterations, for all populations, including those that went extinct (e.g., had a final size of 0);
- The mean final population size for those iterations that do not become extinct, with SE and SD across iterations;
- The final age-sex composition of the extant populations;

- The mean population growth rate, r, with SE and SD across iterations. When harvesting or supplementation are included in your model, VORTEX will report the mean population growth rate for years without harvest or supplementation, for years with harvest or supplementation, and averaged across all years. VORTEX reports the growth rate as the exponential growth (r) rather than the arithmetic growth rate (lambda), because the mean r over time is equal to the long-term r. (Whereas the geometric mean lambda is the long term lambda.) This mean growth rate is calculated from the mean (across years and across iterations) of N in one year over N in the prior year. With the normal sequence of steps in a year, this ratio is calculated before any truncation of N due to exceeding carrying capacity. Thus, if K = 100, N(year 0) = 80, N(year 1 before truncation at K) = 120, the growth rate used in the mean would be $r = \ln(120/80) = 0.405$, even though the growth rate experienced after considering the truncation for K would be $\ln(100/80)$. VORTEX normally reports the growth calculated before truncation for K, because that better reflects the demographic potential for population growth. However, you can change when in the simulated year the growth rate is calculated by changing the position of the rCalc step on the Scenario Settings input page.

Vortex 10 - New Project - C:\Visual Studio 2010\Vortex\Vortex10\bin\Release\New Project\New Project.xml

File Simulation Help
Det. ST

Project Settings Simulation Input Text Output Tables and Graphs Project Report
Input Summary Deterministic Results Output Summary Output Tables ST Tables

Scenario: Default Scenario Population: Population1

Year 100
N[Extinct] = 1, P[E] = 0.010
N[Surviving] = 99, P[S] = 0.990
Mean size (all populations) = 87.24 (1.87 SE; 18.73 SD)
Means across extant populations only:
Population size = 88.12 (1.67 SE; 16.61 SD)
Expected heterozygosity = 0.8069 (0.0049 SE; 0.0486 SD)
Observed heterozygosity = 0.8266 (0.0070 SE; 0.0696 SD)
Number of extant alleles = 8.86 (0.16 SE; 1.64 SD)
Number of mt haplotypes = 2.79 (0.08 SE; 0.84 SD)
Lethal alleles / diploid = 1.37 (0.05 SE; 0.47 SD)

In 100 simulations of Default Scenario for 100 years:
1 went extinct and 99 survived.
This gives a probability of extinction of 0.01000 (0.00995 SE).
or a probability of success of 0.99000 (0.00995 SE).
1 simulations went extinct at least once.
Of those going extinct,
mean time to first extinction was 53.00 years (0.00 SE, 0.00 SD).
Means across all populations (extant and extinct) ...
Mean final population was 87.24 (1.87 SE; 18.73 SD).
Age 0 1 Adults Total
0.00 7.98 36.46 44.44 Males
0.00 8.00 34.80 42.80 Females

Means across extant populations only ...
Mean final N for extant populations was 88.12 (1.67 SE; 16.61 SD).
Age 0 1 Adults Total
0.00 8.06 36.83 44.89 Males
0.00 8.08 35.15 43.23 Females

Across all years, prior to carrying capacity truncation,
mean growth rate (r) was 0.0632 (0.0013 SE; 0.1251 SD)
Final expected heterozygosity was 0.8069 (0.0049 SE; 0.0486 SD)
Final observed heterozygosity was 0.8266 (0.0070 SE; 0.0696 SD)
Final number of alleles was 8.86 (0.16 SE; 1.64 SD)
Final number of mt haplotypes was 2.79 (0.08 SE; 0.84 SD)
Final lethal alleles / diploid was 1.368 (0.047 SE; 0.468 SD)
Mean Ne calculated from loss of heterozygosity in extant populations: 49.79

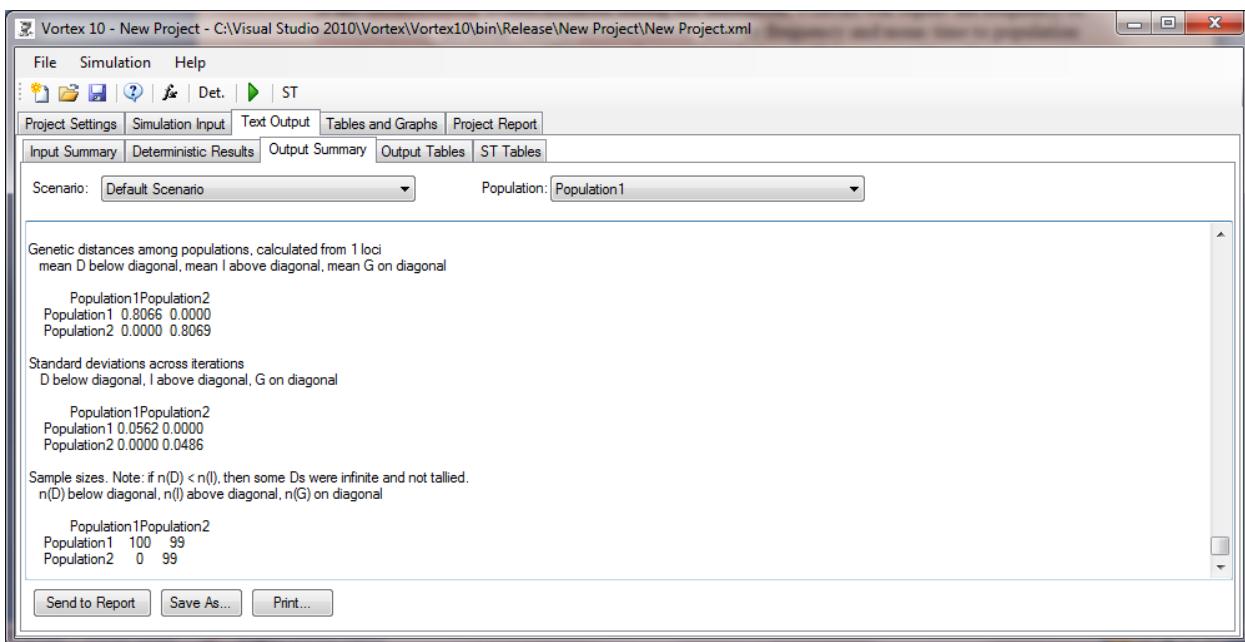
Send to Report Save As... Print...

- The mean effective population size in extant populations, calculated from the loss of gene diversity from year 1 to the last year. Note that this Ne is the size of a randomly breeding population each generation (i.e., tallying only the adults, not the juveniles that constitute the next generation) across the generations. This will be approximately the harmonic mean of Ne at each generation, and will be less than an arithmetic mean Ne across generations if the populations are growing, declining, or fluctuating in size over time.

The standard deviations reported for summary statistics give an indication of the variation that occurred among iterations of the simulation. The standard errors reported for each summary statistic will indicate whether the number of iterations was large enough to give mean results that are sufficiently stable (precise) for your purposes.

Additional summary information will be provided when you have built a metapopulation model. If any recolonization events occurred during the simulation, *VORTEX* will report the frequency of recolonization, the mean time to recolonization, and the frequency and mean time to population re-extinction if appropriate.

Also given for metapopulations will be some tables of genetic distances (and identities), with standard deviations, to show the amount of genetic differentiation that existed between populations at the end of the simulations. The measures of genetic distance that are displayed are Nei's standard genetic distance, D ; genetic identity, I ; gene identity, J ; and Gst . Note that genetic distance, D , is undefined (and will be displayed in the output file as 0, with a sample size of 0) if the populations have no shared alleles because of no dispersal between them. The output also provides the overall Gst among populations, both unweighted and weighted by population size. See Nei (1987) and the box below for more details about these measures of genetic distance.



Genetic distances are normally calculated from allele frequencies, and the matrices of distance given in the Output Summary are calculated from the allele frequencies across the loci specified for summary statistics (see [Genetics](#)). To obtain more precise measures, you can model more neutral loci. You can also observe how pre-existing genetic structure is projected to change over time by specifying starting allele frequencies and then having genetic summary statistics calculated on only those additional loci.

A variable **GDIST(p1;p2)** that gives the genetic distance between populations **p1** and **p2** is available for use in [functions](#) (with **GDIST(p1;p1)** being a code for the genetic distance between **p1** and the metapopulation), so that the genetic distance at each year of the simulation can be obtained by setting a Population State variable to **GDIST(p1;p2)**. Which of the genetic distance metrics is provided in this variable can be set via the [Special Option GDMx](#), with standard genetic distance, D, being the default if the special option is not used. Set x to 1 to have **GDIST** provide genetic identities, I; x = 2 for gene identities, J; and x = 3 for Gst. The value of **GDIST** will be set to -1 if D is undefined because the two populations share no alleles. Caution: values for **GDIST** are updated in the Census of each annual cycle of events, so you might need to put a Census event before the step at which your model needs to use current values for **GDIST**.

Genetic distances can also be calculated from the matrix of kinships that is calculated from the pedigree. The calculations based on kinships will usually be more accurate than those based on allele frequencies, because the kinship calculations are not affected by the random sampling error associated with a small number of loci. However, the calculations based on frequencies of simulated alleles will be faster if the kinship matrices are not otherwise required for (non-recessive model) inbreeding depression. These equivalent genetic distance matrices based on kinships will be included in the [Output Summary](#) if you choose the option under [Genetics](#) to “Calculate genetic distances based on kinships, as well as allele frequencies”. If this option is specified, then values of **GDIST** used in any function will be the values obtained from the kinships, rather than the values estimated from allele frequencies.

How is the genetic distance (or similarity) between populations measured?

A full explanation of the many measures of genetic distance would require a long section of a textbook, and those interested are encouraged to consult Nei (1987), Hedrick (2000), or other texts for detailed derivations. A brief description of the metrics provided in *VORTEX* is given here.

Most measures of genetic distance start with the “expected heterozygosity” – that is, the heterozygosity that would be observed if a diploid population were in hardy-Weinberg equilibrium. Often, this is termed just “heterozygosity”, but that is a bad practice because it can deviate far from actual (observed) heterozygosity, and because the concept can be defined even for species that are not diploid (and, therefore, for which the term heterozygosity is meaningless). A more appropriate term is “gene diversity”, often symbolized G or H, with its converse (1-G) being “gene identity”, often symbolized J. When calculated from allele frequencies (p_i) at a locus, $J = \sum(p_i^2)$ and $G = 1 - \sum(p_i^2)$, with the summation across all alleles at the locus. Across loci, G and J are simply averaged. Averaged across subpopulations, the average gene identity within s subpopulations is $J_s = \sum(J_x)$.

For two populations, the gene identity between them is $J_{xy} = \sum(p_{xi})(p_{yi})$. The total gene identity for a metapopulation J_T can be calculated from overall allele frequencies, and it is also given by $J_T = (\sum J_x + \sum J_{xy})/s^2$, in which the first summation is over all subpopulations, x, and the second summation is over all pairs of subpopulations, in which $x \neq y$.

Converting gene identities to gene diversities (for which I will use the symbol H because it is more common, even if misleading, in the literature):

$$H_T = 1 - J_T \text{ and } H_s = 1 - J_s.$$

The gene diversity between populations, often symbolized D_{ST} , is
 $D_{ST} = H_T - H_s = J_s - J_T$.

With the above background, the most common measures of genetic distance between 2 populations can be defined, as follows:

As noted above, J_{xy} is termed the gene identity between two populations.

Genetic Identity is defined as $I = J_{xy} / (\sum J_x \sum J_y)^{0.5}$, and it scales from 0 to 1 for populations that are completely unique (sharing no alleles) to ones that have identical allele frequencies.

Nei’s “standard genetic distance” is defined as $D = -\ln(I)$, and it scales from 0 to infinity for populations that are identical to ones that share no alleles (note that D becomes undefined when two populations have no alleles in common).

Finally (at least with respect to genetic distance measures reported by *VORTEX*), a very commonly used measure of genetic differentiation is G_{ST} (more commonly symbolized as F_{ST} , although F_{ST} is used in various contexts, not all of which relate directly to allele frequencies).

$$G_{ST} = D_{ST} / H_T = (H_T - H_s) / H_T.$$

Unfortunately, there are a number of problems with G_{ST} as a measure of population divergence, the most notable being that it does not actually measure the genetic difference between populations! G_{ST} measures the reduction in gene diversity (H) within subpopulations relative to the gene diversity of the metapopulation. If the subpopulations each have high diversity, then G_{ST} will be nearly 0 even if the subpopulations share almost no alleles. In fact, distinct species can have $G_{ST} \approx 0$, so obviously it is not a good measure of genetic divergence. The history of G_{ST} (or F_{ST}) being used as a measure of population differentiation arises because when there are only two alleles in the metapopulation at each locus studied, then populations cannot become divergent without also losing gene diversity (becoming more homozygous), as they drift to fixation for the alternate alleles. Thus, in this special case, within-population gene identity becomes highly correlated with (and thus an acceptable indirect measure of) divergence.

How can pedigree kinships be used to calculate genetic distance between populations measured?

The methods described in the above box are used to quantify the genetic similarity or divergence between populations. Analogous methods can be used to estimate the same quantities from the kinship coefficients that can be calculated from any pedigree.

First, note that the coefficient of kinship (also termed the coefficient of consanguinity) is defined as the probability that two homologous alleles are identical by descent (ibd). By definition, the alleles in the founders of a pedigree are not identical by descent, so analysis of probabilities of alleles being shared ibd is the same as an analysis of an “infinite alleles” model of genetic loci, in which each founder contributes initially two unique alleles to the population. (This is why *VORTEX* uses just such an infinite alleles model of transmission of founder alleles as its default for the simulation.) Thus, the initial allele frequencies are all $p = 1/(2N_f)$, in which N_f is the number of founders. In subsequent generations descended from these founders, gene identity within the population is $J = \sum(p_i^2)$ and gene diversity is $G = 1 - \sum(p_i^2)$.

But note that since the kinship between any two individuals is the probability that two alleles sampled from them are ibd, and thus the mean of all pairwise kinships within the population (which can be symbolized as MK_w) is the probability that two randomly sampled homologous alleles will be ibd – i.e., $MK_w = J$. Moreover, for comparisons between two populations x and y , the mean across all kinships of an individual in population x to an individual in population y (symbolized MK_b , for the mean kinship between the populations) will be the probability that homologous alleles sampled one from x and one from y are ibd – thus, $MK_b = \sum(p_{xi})(p_{yi}) = J_{xy}$. Similarly, J_T is the mean of all pairwise kinships in the metapopulation.

Therefore, all the gene identity components that are needed for calculating the standard measures of genetic distance among subpopulations can be derived from the matrix of pairwise kinships, defining the relationships among all individuals in the metapopulation, and its submatrices extracted for each subpopulation. A convenient aspect of using the kinship matrix to calculate the measures of genetic differentiation in a metapopulation is that since the kinship of each individual to every other individual in the metapopulation is known, it is a simple matter to determine how the diversity within each population, and between populations would change if any individuals were moved into other populations. This can provide a powerful method for determining the optimal number, timing, and even individuals for translocation to achieve genetic management goals for a metapopulation (such as keeping within-population gene diversity sufficiently high, or maintaining between-population divergence at a desired – low or high – level).

The choice between using allele frequency data versus pedigree data to calculate genetic distance metrics is a matter of what kind of data are available. In captive and other intensively managed populations for which the pedigree can be tracked through the generations, the kinship methods are easily applied and give precise values based on probabilities of shared alleles. When the pedigree is not known, assays of genotypes (e.g., from allozymes, microsatellite DNA, SNPs, or full sequences) can be used to quantify genetic structure via the methods based on allele frequencies. (But note that some of the measures, such as G_{ST} , yield results that are very highly dependent on which measure of genetic diversity is used to obtain allele frequencies.) When both allele frequency data and pedigree relationships are available, either methodology (or both) can be used, and they should give comparable results (unless the allele frequency data are inconsistent with the recorded pedigree). In simulation models, such as *VORTEX*, the data are all hypothetical, so we can use either method and make the results as precise as we want!

Output Tables

The fourth section of Text Output provides three kinds of summary tables. The Scenario Summaries provides a line of basic summary statistics for each Population of each Scenario that has been run. The summary statistics tabulated are the number of iterations (#Runs), the deterministic growth rate (det-r), the mean stochastic growth rate (stoch-r) experienced in the simulations, the SD of the stochastic population growth [SD(r)], and final values (at the end of the simulation) for many of the descriptive statistics listed above in the Output Summary. The Scenario Summaries tables also list the mean and SD at the end of the simulation for any Global State variables and Population State variables. Note that the column headers for these means and SDs are shown as GSvars, SD(GSvars), PSvars, and SD(PSvars), regardless of how many GSvars and PSvars are in your projects. The header simply indicates that the last sets of columns show means and SDs of the state variables, but the actual number of state variables in your project cannot be shown in the column headers because different scenarios often have different numbers of state variables. The means of Population State variables listed for a metapopulation (when there is more than 1 population) will be the mean across the populations of their PS variable means.

The screenshot shows the 'Output Tables' window in Vortex 10. The window title is 'Vortex 10 - New Project - C:\Vortex10Projects\New Project\New Project.xml'. The menu bar includes File, Simulation, Help, Det., and ST. The tabs at the top are Project Settings, Simulation Input, Text Output, Tables and Graphs, and Project Report. The 'Text Output' tab is selected. Below the tabs are sub-tabs: Input Summary, Deterministic Results, Output Summary, Output Tables, and ST Tables. The 'Output Tables' sub-tab is selected. The main area contains a table with the following columns: Scenario, #Runs, Population, det-r, stoch-r, SD(r), PE, N-extant, SD(Next), N-all, SD(Nall), GeneDiv, SD(GD), AllelN, SD(A), MedianTE, MeanTE, and StateVars. The table data is organized by scenario (Default Scenario, Default Scenario-Copy, NewDefault, NewMK) and population (Population1, Population2, Population3, Population4, Meta). Each row provides detailed statistics for each iteration of extinction or survival, including final population sizes, gene diversity, and inbreeding coefficients.

The Iteration Summaries in Output Tables provides a tabulation for each iteration of the year of extinction (if extinction occurs), final population size, and, if extinction does not occur, the final gene diversity, mean inbreeding coefficient, and number of alleles remaining. The data in this table can be used to analyze the full distribution (not just the mean and SD) of the times to extinction, final population sizes, and genetic statistics. The “Save As” button will export the tables as text files delimited with semi-colons or commas, or as Excel files.

Vortex 10 - New Project - C:\Vortex10Projects\New Project\New Project.xml

Scenario: NewMK

| Iteration | YrExt | Population1 | | | | Population2 | | | | Population3 | | | | Population4 | | | | Metapopulation | | | |
|-----------|-------|-------------|---------|--------|---------|-------------|---------|--------|---------|-------------|---------|--------|---------|-------------|---------|--------|---------|----------------|--------|---------|--------|
| | | N | GeneDiv | Inbred | Alleles | YrExt | N | GeneDiv | Inbred |
| 1 | 78 | 0.9003 | 0.1026 | 19.00 | 88 | 0.9338 | 0.0000 | 21.00 | 98 | 0.8879 | 0.1224 | 17.00 | 60 | 0.8103 | 0.1667 | 8.00 | | 324 | 0.9506 | 0.092 | |
| 2 | 85 | 0.9166 | 0.0824 | 24.00 | 94 | 0.9388 | 0.0638 | 21.00 | 88 | 0.8844 | 0.0795 | 21.00 | 98 | 0.8405 | 0.1122 | 11.00 | | 365 | 0.9543 | 0.084 | |
| 3 | 78 | 0.8895 | 0.1538 | 17.00 | 84 | 0.8654 | 0.1071 | 17.00 | 98 | 0.8946 | 0.1224 | 16.00 | 89 | 0.7892 | 0.1573 | 9.00 | | 349 | 0.9302 | 0.134 | |
| 4 | 88 | 0.8840 | 0.0909 | 19.00 | 53 | 0.9293 | 0.1321 | 20.00 | 76 | 0.9033 | 0.0921 | 19.00 | 97 | 0.6923 | 0.2887 | 6.00 | | 314 | 0.9375 | 0.159 | |
| 5 | 94 | 0.9303 | 0.0745 | 23.00 | 97 | 0.9178 | 0.0619 | 23.00 | 79 | 0.9103 | 0.0506 | 18.00 | 95 | 0.7812 | 0.2105 | 10.00 | | 365 | 0.9511 | 0.101 | |
| 6 | 98 | 0.9212 | 0.0816 | 21.00 | 90 | 0.9321 | 0.0556 | 27.00 | 99 | 0.9321 | 0.0202 | 23.00 | 99 | 0.8370 | 0.0909 | 9.00 | | 386 | 0.9605 | 0.062 | |
| 7 | 94 | 0.8511 | 0.1596 | 13.00 | 84 | 0.8855 | 0.0952 | 17.00 | 89 | 0.9063 | 0.1461 | 17.00 | 88 | 0.8499 | 0.1023 | 8.00 | | 355 | 0.9332 | 0.126 | |
| 8 | 93 | 0.9209 | 0.0968 | 22.00 | 92 | 0.9340 | 0.0435 | 26.00 | 98 | 0.9232 | 0.0408 | 23.00 | 101 | 0.7778 | 0.1683 | 6.00 | | 384 | 0.9536 | 0.088 | |
| 9 | 80 | 0.8748 | 0.0875 | 18.00 | 92 | 0.9146 | 0.0870 | 20.00 | 101 | 0.8913 | 0.0891 | 19.00 | 99 | 0.7954 | 0.2323 | 11.00 | | 372 | 0.9414 | 0.126 | |
| 10 | 87 | 0.9139 | 0.1264 | 19.00 | 86 | 0.9192 | 0.0581 | 17.00 | 95 | 0.9076 | 0.1158 | 19.00 | 80 | 0.8654 | 0.1125 | 11.00 | | 348 | 0.9547 | 0.103 | |
| 11 | 97 | 0.8928 | 0.1031 | 16.00 | 99 | 0.9145 | 0.0808 | 21.00 | 97 | 0.8740 | 0.0412 | 18.00 | 100 | 0.8101 | 0.2500 | 7.00 | | 393 | 0.9427 | 0.119 | |
| 12 | 100 | 0.9211 | 0.0900 | 22.00 | 96 | 0.9278 | 0.0417 | 24.00 | 90 | 0.8857 | 0.1000 | 19.00 | 99 | 0.7804 | 0.2121 | 9.00 | | 385 | 0.9453 | 0.111 | |
| 13 | 100 | 0.8491 | 0.1500 | 18.00 | 82 | 0.9015 | 0.0854 | 18.00 | 99 | 0.9143 | 0.0909 | 19.00 | 101 | 0.8445 | 0.0990 | 10.00 | | 382 | 0.9383 | 0.107 | |
| 14 | 91 | 0.8933 | 0.0769 | 19.00 | 60 | 0.9308 | 0.0833 | 22.00 | 73 | 0.9202 | 0.0274 | 16.00 | 73 | 0.8550 | 0.0822 | 11.00 | | 297 | 0.9549 | 0.067 | |
| 15 | 98 | 0.8510 | 0.1531 | 18.00 | 93 | 0.8530 | 0.0968 | 20.00 | 100 | 0.8097 | 0.1400 | 17.00 | 89 | 0.6446 | 0.3708 | 6.00 | | 380 | 0.8941 | 0.186 | |
| 16 | 93 | 0.9181 | 0.0753 | 20.00 | 98 | 0.9214 | 0.0510 | 24.00 | 99 | 0.9266 | 0.0505 | 21.00 | 95 | 0.8189 | 0.1368 | 8.00 | | 385 | 0.9550 | 0.077 | |
| 17 | 100 | 0.9330 | 0.0600 | 24.00 | 83 | 0.9352 | 0.0602 | 22.00 | 94 | 0.8940 | 0.0957 | 19.00 | 103 | 0.8501 | 0.0971 | 10.00 | | 380 | 0.9570 | 0.078 | |

Save As... Print...

If more than one population is modeled, then the Metapopulation Structure in Output Tables provides in tabular format the various measures of population divergence that are included at the bottom of the Output Summary text. You need to choose for which scenario you want to see these statistics and which of a number of measures of divergence to display. See the boxes above (in [Output Summary](#)) for more explanation of these genetic metrics. If you choose the [Genetics](#) option to “Calculate genetic distances based on kinships, as well as allele frequencies”, then these additional measures based on kinships will be included in the drop-down list of the genetic metrics that you can display.

Vortex 10 - New Project - C:\Vortex10Projects\New Project\New Project.xml

Scenario: NewMK

Genetic metric to display: I (Genetic Identity) below, SD above, Jx on diagonal

| | Population1 | Population2 | Population3 | Population4 |
|-------------|-------------|-------------|-------------|-------------|
| Population1 | 0.0995 | 0.1012 | 0.1340 | 0.0000 |
| Population2 | 0.7277 | 0.0872 | 0.1009 | 0.0000 |
| Population3 | 0.5342 | 0.7260 | 0.0996 | 0.0000 |
| Population4 | 0.0000 | 0.0000 | 0.0000 | 0.1994 |

Save As... Print...

ST Tables

The last tabbed page of Text Output provides the same information as the Scenario Summaries of the Output Tables, but with results from ST Analyses. (This tab will be blank if no Sensitivity Tests have been run.)

Other Output Files

Another standard output file (with extension .N) provides a tally by year and iteration of the population size. These are the data that are used to create the display of N while the simulations are running. There are several other optional output files: See [Special Options](#) under Project Settings, above. Note that the .N file and the optional output files can be very large, but they also provide the raw data that can be fed into other programs for further statistical analysis.

Tables and Graphs

This section provides quick access to 6 standard graphs of simulation results, as well as the option for Custom Graphs that provides an array of output variables for graphing.

The populations to be graphed for the selected scenarios are specified via a simple list of checkboxes. All years up to an optional user-specified limit are shown on the graphs and tables.

Lines labels (as shown in the graph legend and on the table) can now be edited from a table that is accessed from an “Edit Line Labels” button below the graph. After editing, you must either Accept or Cancel your changes.

Sometimes changes to graph settings will not immediately result in a changed graph. (Usually this happens because the cursor is still on the changed setting, so the program has not yet accepted and processed the change that is being made.) In such cases, the graph can be updated with the current settings by clicking the “Update Plot” button.

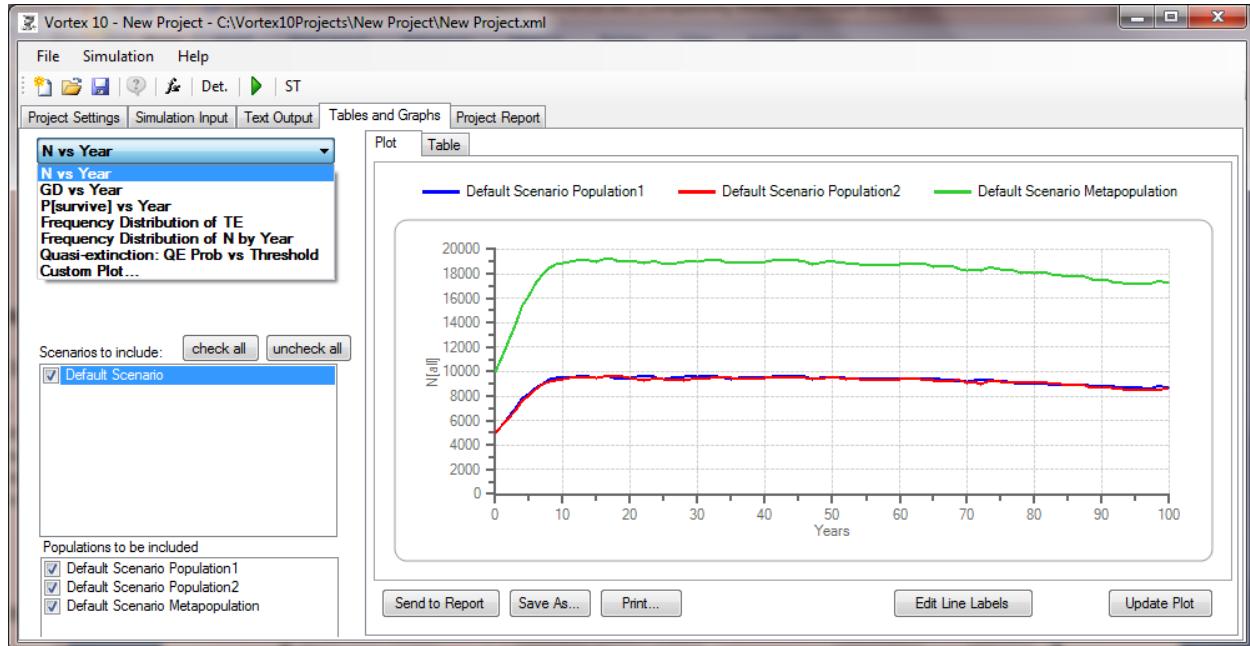
When any graph is displayed, the data used in creating that graph can be viewed (and optionally saved or printed) by going to the Table tab.

Double-clicking on a graph will open the Chart Properties, in which you can change any of the settings (and even change the data!) for the graph.

Note: the Tables & Graphs tab will not be visible if you have not yet run any scenarios, because there are no data to be shown in the Tables & Graphs!

Standard Graphs

N vs Year – the same as the N[all] graph in the custom graphs.

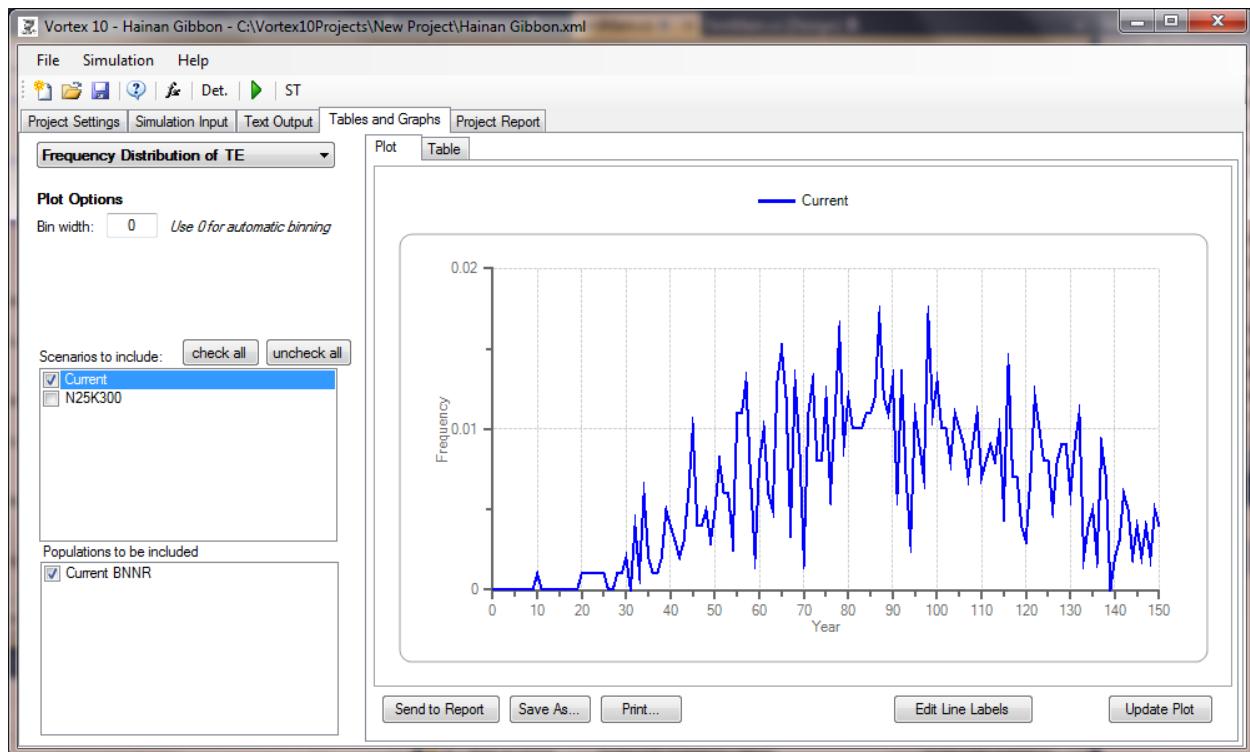


GD vs Year – same as the Gene Div graph in the custom graphs.

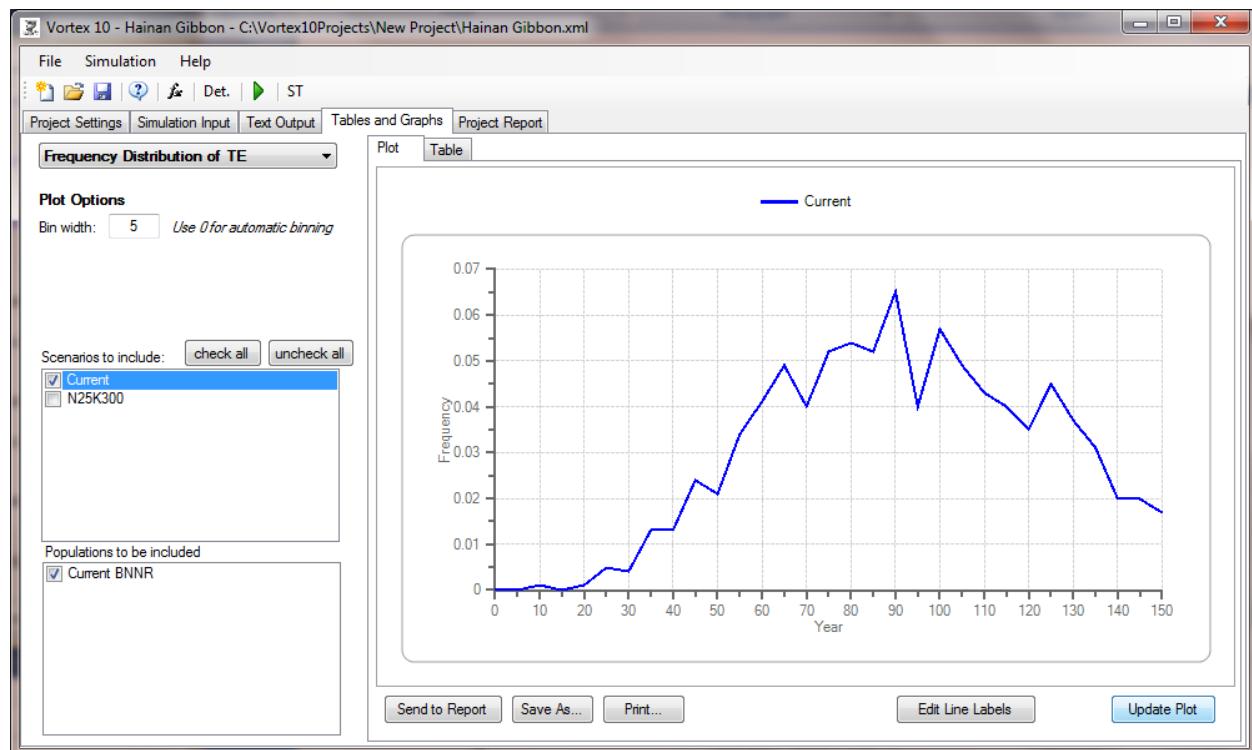
PE vs Year – same as the P[survive] graph in custom graphs: the probability that a population will be extant at each year.

Frequency Distribution of TE – proportion of iterations becoming extinct at each year. This graph will be blank if there have been no extinctions.

The bin width determines how smooth the graph will be. If a lot of iterations were run, and many resulted in extinction, then a narrow bin width (even 1) will show the frequency distribution in the finest detail. Wider bins combine counts across years. If you enter 0 for bin width, then VORTEX will analyze your data to try to determine a good bin width for presenting your results. Often it does a good job, but sometimes it does not, and you can always test alternative bin widths to see what resolution of the distribution looks best to you.

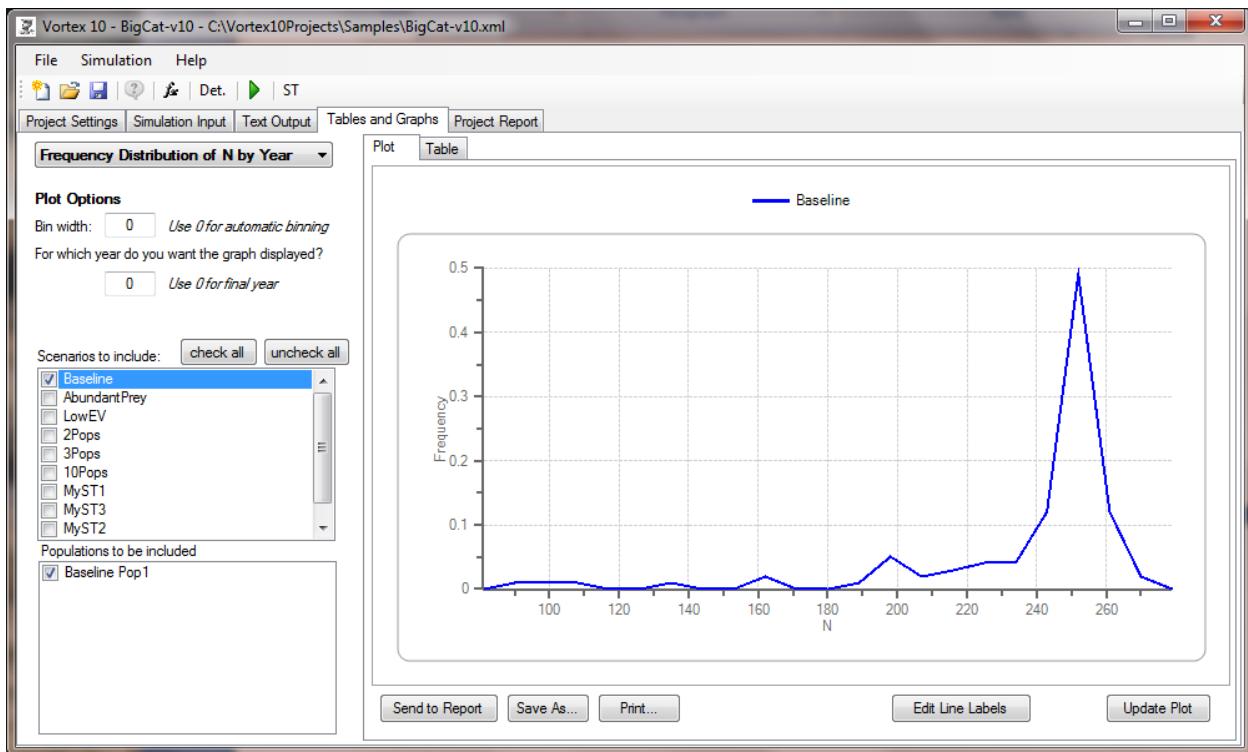


A frequency distribution of time to extinction, with the default Bin Width of 1 year.

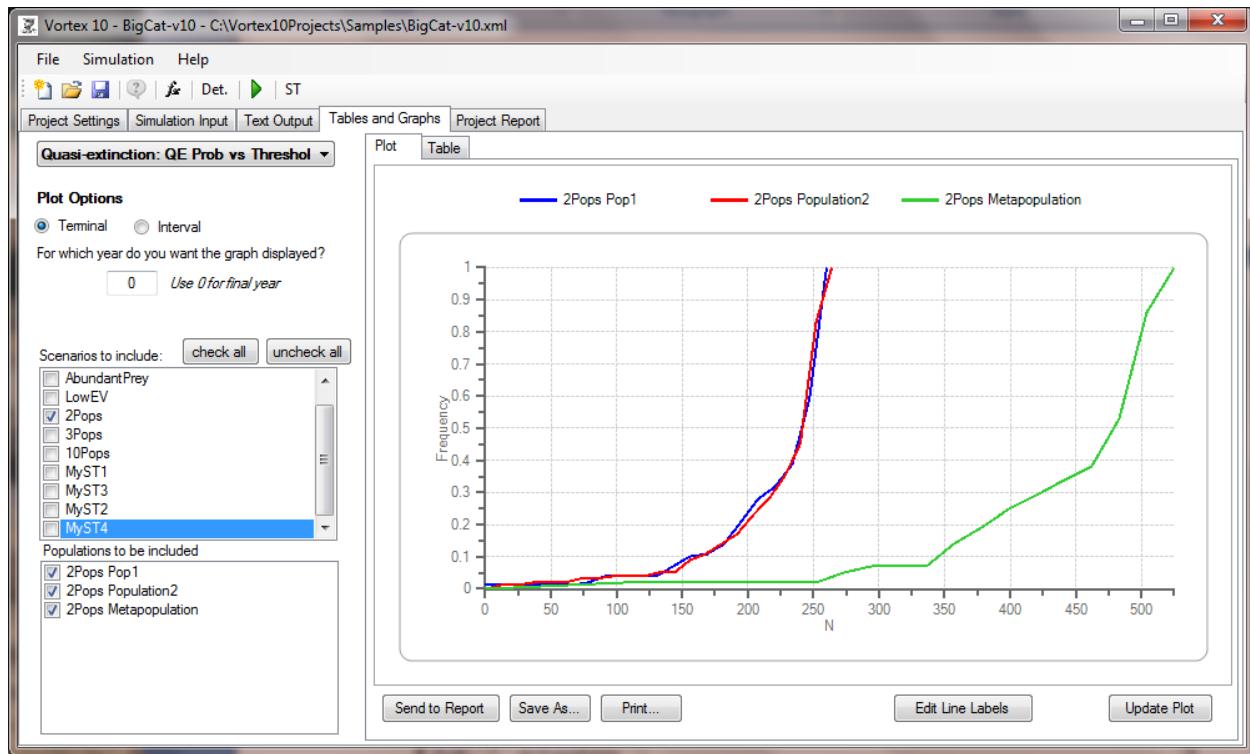


The same Frequency Distribution of TE but with Bin Width set to 5 years.

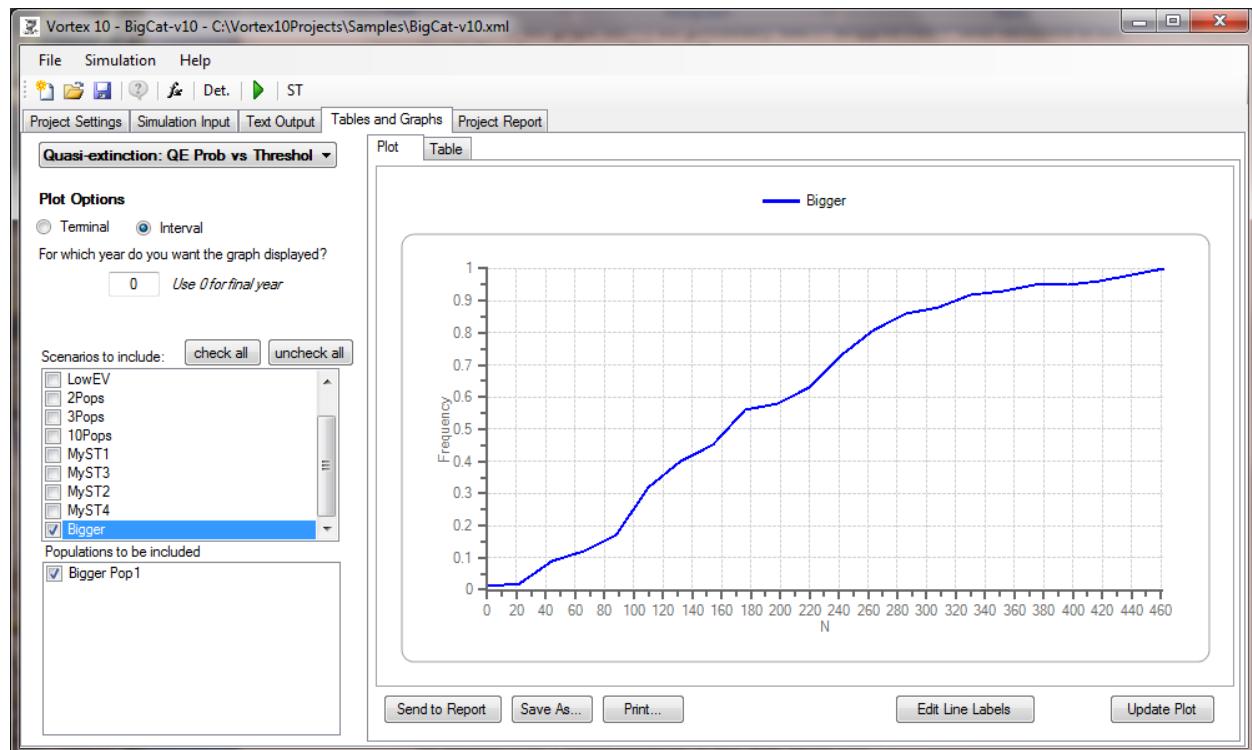
Frequency Distribution of N by Year – Proportion of iterations having each population size as of the specified year (default is the last year). You can set the bin to a value greater than 1 so as to smooth the frequency distribution. This type of graph and Quasi-Extinction graph (see below) also provide an option to specify that the year for which the results are displayed is sometime before the final year of the simulation (which is the default year for these graphs). This can be useful when, for example, you run a simulation for 200 years but you want to examine the distribution of N or the extinction (or quasi-extinction) probabilities also at perhaps year 50, year 100, and year 150.



Quasi-extinction: QE Prob vs Threshold – The probability that the population size was below each possible N. If the “Terminal” option is chosen, then the graph shows the probability of N being below each threshold at the end year (for this graph, which can be set to a year prior to the final year of the simulation). If the “Interval” option is chosen, then the graph shows the probability that N dropped below each threshold at any time before the end year of the graph.



The quasi-extinction graphs can be used to determine the likelihood of “extinction” for any population size that is chosen as the definition of extinction. This can be useful if you want to know the likelihood that the population will fall below some goal that might be set for management reasons or because other analyses suggest that such a population size is a threshold below which population dynamics enter an unstable or otherwise undesirable zone. An application that might be especially valuable is in the determination of threatened status and setting recovery goals for species (Carroll et al. 2014). The US Endangered Species Act defines a Threatened species as one that is likely to become Endangered in the foreseeable future. If Endangered for a species is specified, for example, to be a more than 20% chance of extinction within 100 years, then sensitivity testing might indicate that Endangered should be defined to be a population with N less than – for example – 160. Interval quasi-extinction graphs examined for simulations with different starting populations sizes might then indicate that a population with initial N = 500 has a 50% chance of falling below this Endangered threshold within the next 100 years (as in the next Figure), so that criterion defining Threatened status (or, the reverse, the delisting goal for exiting from Threatened status) would then be set at N = 500. This could be a rigorous method to define the threshold for the Threatened status in a way that is consistent with the wording of the law (as opposed to the wild guess hand-waving that is usually done now, when any quantitative definition is given at all!).

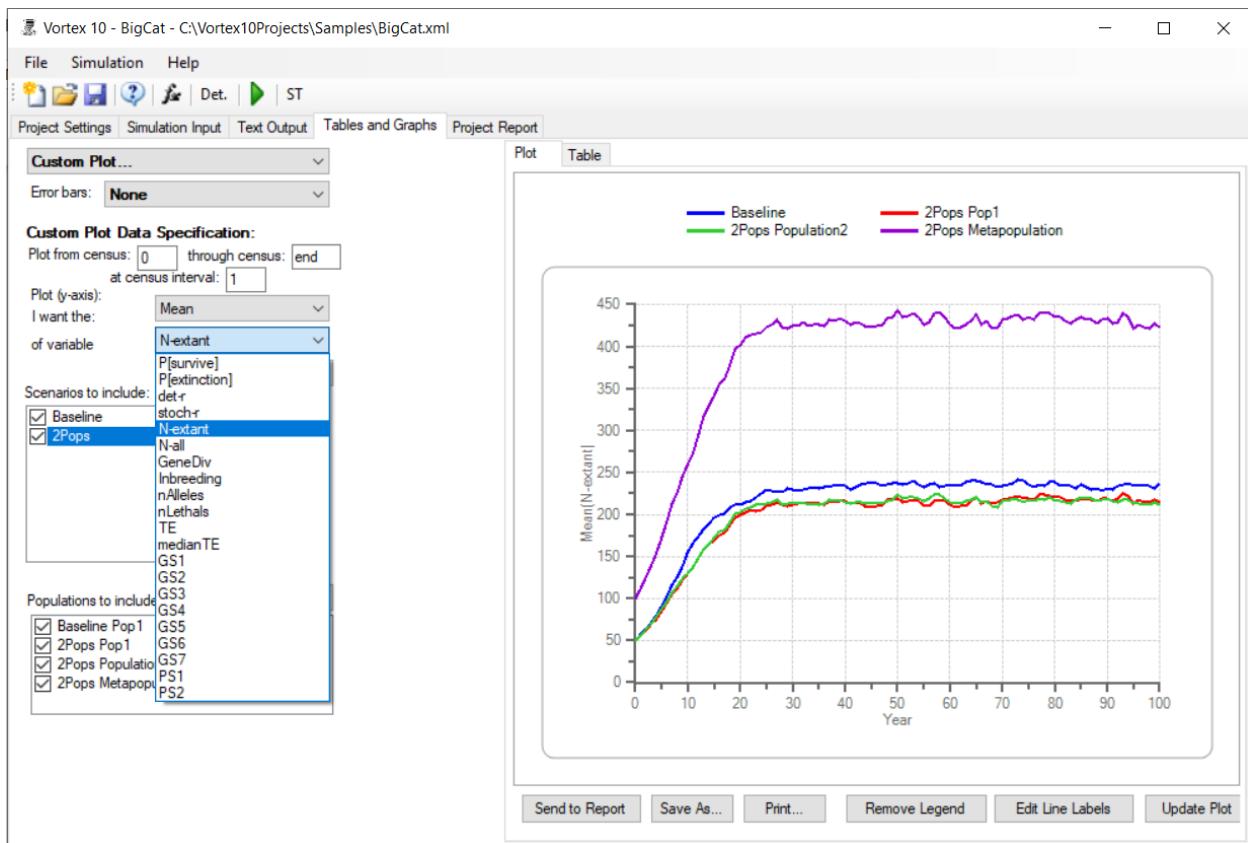


Custom Plots

Selecting “Custom plots …” lets you pick any of a number of output variables to be plotted against year. Variables available for plotting are:

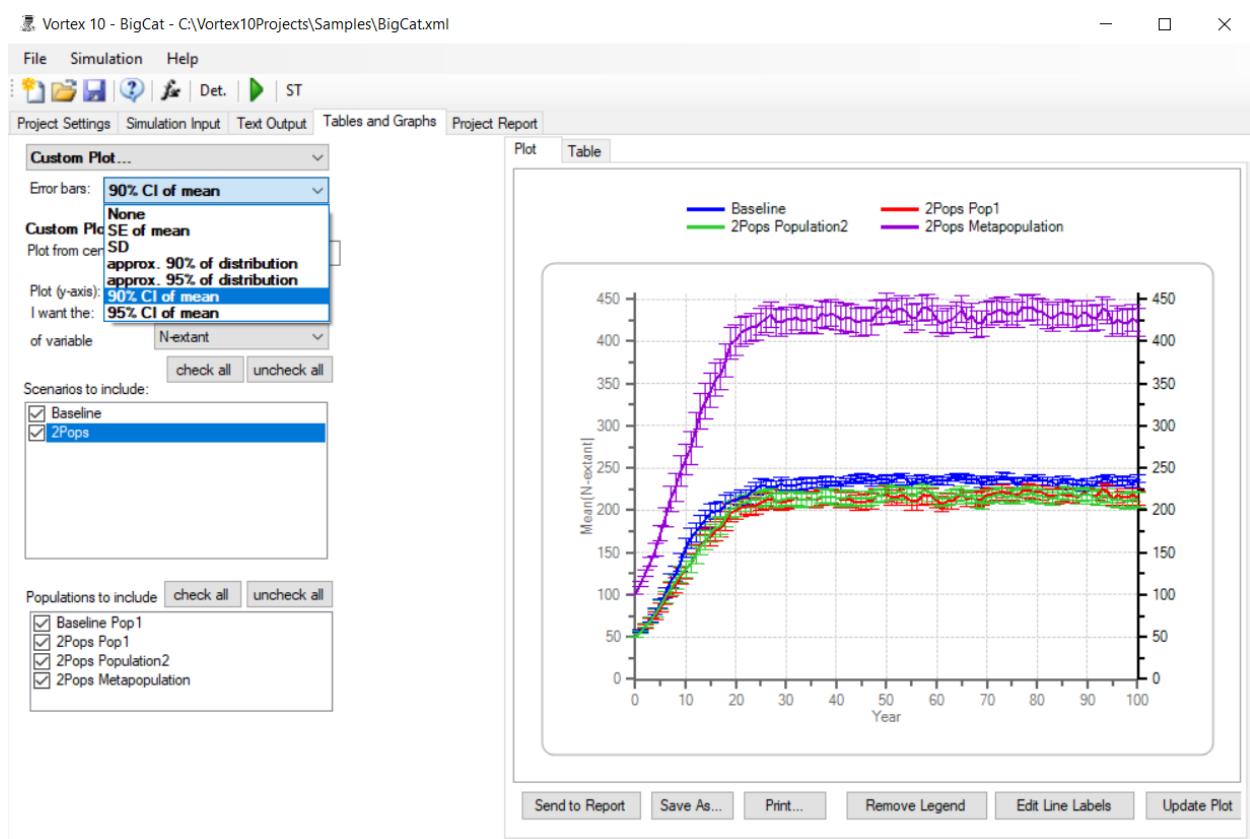
- P[survive] – The proportion of iterations in which the population is not extinct
- P[extinction] – The probability that the population becomes extinct (PE) that year
- Det r – The deterministic growth rate calculated from mean birth and death rates
- Stoch r – The mean population growth rate experienced in the simulations, averaged across all years in which the population was extant. Note that with the default sequence of events in a year, stochastic r will be calculated before carrying capacity truncation.
- N[all] – The mean population size, across all populations extant or extinct
- N[extant] – The mean population size, only for those remaining extant
- GeneDiv – The mean “gene diversity” (or “expected heterozygosity”) remaining in the extant populations.
- Inbreeding – 1 - mean “observed heterozygosity” remaining in the extant populations
- nAlleles – The mean number of alleles remaining within extant populations (from an original number equal to twice the number of founder individuals)
- nLethals – The mean number of lethal alleles remaining within extant populations
- TE – Of iterations that suffer extinctions, the mean time to first population extinction
- Median TE – If at least 50% of the iterations went extinct, the median time to extinction

Along standard output variables, you have the option of selecting any of the GSvars or PSvars that were in your scenario.



You can plot from Year 0 up to any year of the simulations, although the default will be that you see the full span of years. You also specify if you want a plot of the Mean (the default selection), or Standard Deviation, or Standard Error. If you choose Mean, then you can add error bars showing the SD, SE of the mean, 95% or 90% confidence interval around the means, or 95% or 90% range of the distribution of results from individual runs. The 90% and 95% of the distribution of runs can be used to compare, for example, a single observed (real) trajectory to the range of trajectories generated by the simulation – to see if the observed population trajectory would be an outlier among (and therefore perhaps poorly modeled by) the simulations. Both the CIs and the distribution percentiles are approximate, as they assume a symmetrical normal distribution of the results. More precise calculations of errors and percentiles of individual results could be obtained by analyzing the full distribution of simulation results. (See next figure.) You also need to specify for which variable you want to see the results plotted.

For all of the graphs (and almost all graphs displayed in any part of *VORTEX*), you have the options to Send to Report (which also leaves an image of the graph in the Windows clipboard, available for pasting into Word, PowerPoint, or other documents), Save As (which opens a dialog box to let you select the image format in which you want the graph saved), or Print. You can also hit “Edit Line Labels” to open up a list of the labels for editing. (The default labels that *VORTEX* creates are combinations of Scenario and Population names, and can be rather long and cumbersome.)

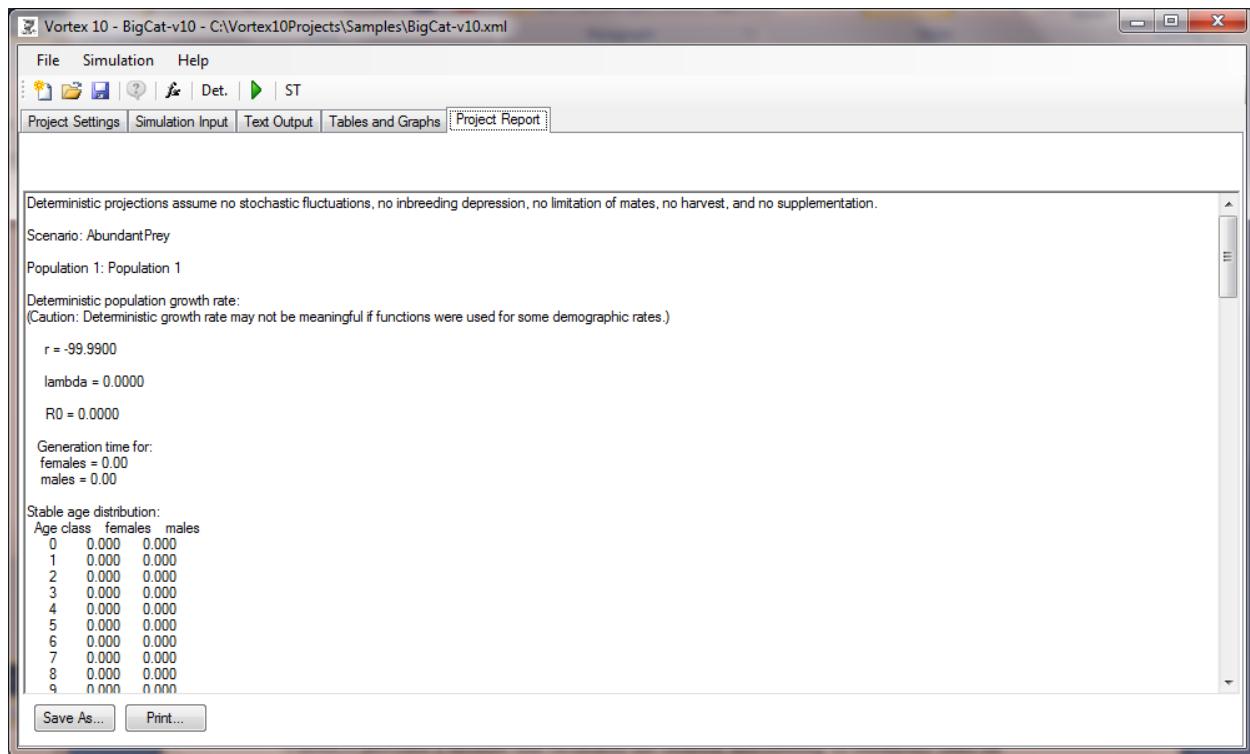


Remember that after changing options for a graph, you sometimes will need to hit the “Update Plot” button to get your new options to be shown in the graph.

Project Report

This is a basic (Notepad-like) text Window in which you can build your own Report. Text, tables, and graphs from other screens can often be sent to the Report, and you can directly edit the Report as well. You can cut and paste from other data fields (e.g., from your Project or Input Section Notes) into the Project Report, and you can cut and paste between the Project Report and other Windows documents.

In Text Output, Tables and Graphs, and on several other screens, you had the option to send text, tables, and graphs to your Project Report. This is where you were sending them! It is always a good idea to liberally send information to your Project Report whenever you think that it may be information that you will want to capture for inclusion in project reports of any sort. It is always easy later to edit or delete sections of your Project Report, but it may be difficult to later resurrect information that you neglected to send to the report.



The Project Report will automatically be saved as an rtf file in the Project Folder when you save your Project, and it will automatically re-load the Project Report when you open a project. You also have the option to save a copy of it as an rtf or txt file in any folder with any filename. Such files can be imported directly into MS Word and most other word processors, where you can use their more powerful editing capabilities to further refine your report. You can also Print the Report.

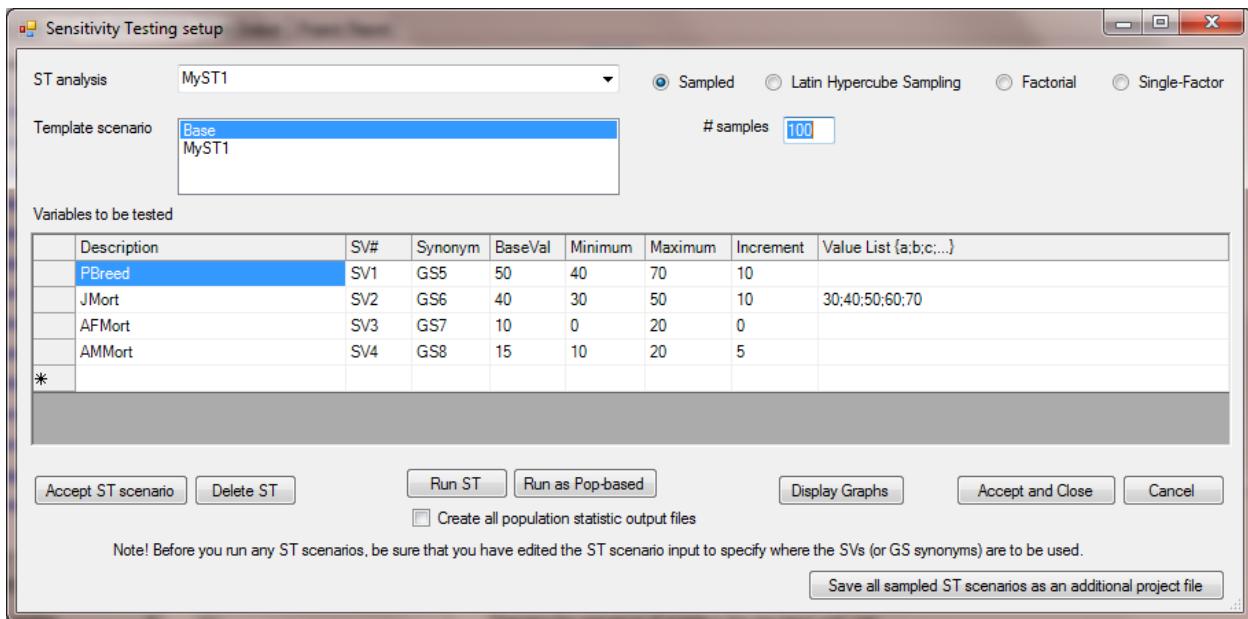
Sensitivity Tests (ST)

Sensitivity Tests are handled in a very different way in *VORTEX* 10 than in *VORTEX* 9, so that they are both more flexible and powerful but also easier to use. Therefore, and STs defined in a *VORTEX* 9 project file will be ignored when the old vpj file is loaded into *VORTEX* 10.

VORTEX provides a module that facilitates the creation and running of sensitivity tests on parameters that are uncertain or subject to management. The method used by *VORTEX* for ST is to create a series of scenarios that explore the uncertain parameter space. These scenarios all have the same values for the input, but the parameters to be tested will be defined to be functions of Scenario number (variable SCENE in the *VORTEX* functions), so that they take on alternative values when evaluated in the cloned scenarios. *VORTEX* first makes a base scenario (“STbase”) by making a copy of a “template” that is one of the scenarios already existing in the project. The STbase is then cloned to create the series of scenarios that are run to complete the ST. The STbase scenario can be (and usually must be) edited to make it properly represent the sensitivity tests to be conducted (see below). However, do not change the name of the STbase scenario after creating the ST; otherwise, *VORTEX* will not be able to run the ST scenarios created from it.

After selecting the template and creating the STbase from it, the user specifies in an STsetup window the parameters and their ranges to be explored. *VORTEX* then creates within STbase some new global state variables with labels SV1, SV2, etc. (for sensitivity variable), defined with functions that specify that the values will vary over the range to be tested. The user then must edit the STbase scenario to place these SVs into values or functions specifying the parameters to be examined with the ST.

You can create multiple STs associated with a *VORTEX* project, and each ST can specify any number of parameters that will be varied. Multiple parameters would be varied within one ST if the desire is to test each value against the range of values for the other parameters. (I.e., main effects and all interactions among the variables can be tested.) If the desire is to test different uncertain parameters (or sets of parameters) one at a time, while holding all other parameters constant, then those separate tests should be placed into separate STs in the *VORTEX* project.



Much of the mechanics and values of the ST module are more easily explained by stepping through the use of the tool. The screen shot below shows the STsetup window. The steps in using it are as follows:

Creating ST scenarios

1. The first step to run sensitivity tests on parameters in a *VORTEX* project is to provide a name for the ST in the “ST analysis” text box at the top of the window. Do not use the default label “New ST” (i.e., change it to something else), so as to not confuse an existing ST with a request for creation a new one.
2. A “Template scenario” is then chosen from the existing scenarios in the project. This template defines all the parameters that will not be varied in the ST. (The template scenario, as a fully functional *VORTEX* scenario, defines also values for those parameters that will be varied, but those values will be over-written by the values tested within the ST.) The template will be copied to a new scenario (which becomes the STbase) that has the same name as given in step #1 to the ST analysis.
3. The ST module provides four options for how the ranges of parameter values will be explored: Sampled, Latin Hypercube Sampling, Factorial, or Single-Factor.
 - a. If “Sampled” is chosen, then for each scenario created in the series for the ST, a value is randomly chosen for each of the tested parameters from across the range specified. With a large number of samples, the parameter space can be thoroughly explored, and statistical analysis (such as a multiple regression) can be used to test for impacts of each parameter and all combinations.
 - b. Latin Hypercube Sampling (LHS) is a method of sampling that more evenly covers the parameter space and thus is more efficient than random sampling. The number of samples (S) to be obtained from the n-dimensional space for n parameters and the range for each parameter are specified. The algorithm then

divides each parameter range evenly to obtain its S values. Sampling then proceeds by the random selection of a value from for each parameter, with this sampling repeated S times under the constraint that each of the S values of each parameter is chosen once. An advantage of LHS is that the n-dimensional space can be represented well with a manageable number of samples, and the number of samples required to explore the uncertainty in each parameter does not increase with the number of parameters.

- c. With the Factorial option, every combination of parameter values is tested, and thus the range of values for each parameter must be a discrete set, not a continuous distribution. This option provides the greatest statistical power for testing the effects of each parameter value and the interactions among parameters, but the total number of tests can become very, very large with even a moderate number of parameters to be tested.
 - d. With the Single-Factor option, each parameter is varied across its range while holding all the other parameters at the base values. This option provides a way to test each parameter independently (usually with far fewer samples than is required for testing the combinations of parameters using one of the first three methods), but it does not allow for testing interactions among parameters. It might be a good option for initial sensitivity tests to determine which factors have a large enough impact to be worth exploring in more detail.
4. For either Sampled or LHS, the number of samples to be taken of the parameter space is specified. For initial tests, either to see if the ST is working or to begin to determine which parameters have large impact and are worth further study, 100 samples might be adequate. For good statistical results, however, often many more should be used – perhaps 1000 or more if there 4 or more parameters being varied. Note, however, that there will be a trade-off with respect to the time it takes to complete the analysis. The total time will be approximately proportional to the #samples in the ST x #iterations in each scenario. Some good statistician can probably provide guidance on whether it is better to increase samples or increase iterations, although the answer likely depends on the output measure that is being assessed (e.g., Prob Extinction requires many iterations to be estimated adequately, while mean N usually requires fewer, and mean GD yet fewer as the latter measures often vary less among iterations).
- a. With the Factorial option, the #samples created in the factorial ST is the product of the number of levels to be tested for each parameter, and the box for the #samples is disabled.
 - b. The #samples has a slightly different meaning for Single-Factor analyses. First, if you specify a discrete set of parameter values to be tested, either with an Increment or by providing a Value List (see below), then the #samples doesn't do anything in a Single-Factor analysis because the number of samples for each factor will just be the number of values in the list. However, if you just give a Minimum and Maximum, with no Increment to define a discrete list, then the #samples for a single-factor analysis will be the number of samples to be taken within that range for each factor. Thus, the #samples applies to each variable that does not have a discrete list, rather than (as in “Sampled” and “LHS” methods) the total number of samples tested in the analysis.

5. In the “Variables to be tested” you specify a SV (which will be made as a synonym for a newly created GSvar) for each parameter that you wish to vary. Note that these SVs will later need to be directly inserted into a *VORTEX* input parameter (e.g., %FemalesBreeding can be set to =SV1) or can be used within a larger function so that a set of *VORTEX* input values are changed synchronously (e.g., if SV2 is to be a multiplier for baseline mortality rates, then each mortality rate might be specified with a function such as =SV2*20, with 20 being the baseline rate for that age-sex class).
 - a. Give each SV a Description that is meaningful to you.
 - b. The synonym is filled in automatically by *VORTEX*, and it specifies to what new GSvar the SV will be assigned. (You can then use either SV1 or the synonymous GS1 in your inputs.) You cannot change either the SV# or the GS synonym, as these are automatically assigned by *VORTEX*.
 - c. The Base Value will not impact the ST itself, but it is useful because you can later run the ST scenario with the base values to provide a baseline set of results.
 - d. There are several methods for specifying the set of parameter values to be tested, and they will be applied in the order specified below:
 - i. If a Value List is provided, then the values are sampled from that list. This should be a list of values separated by semi-colons (not commas!).
 - ii. If no Value List is provided, but a Minimum, Maximum, and Increment (> 0) are all provided, then the values are sampled from the set of discrete values from the minimum to maximum (inclusive), at the specified intervals.
 - iii. However, with LHS, the Increment is determined by the #samples specified. The Increment and the Value List will be calculated and filled in automatically, and they cannot be edited by the user. (If you change the Minimum, Maximum, or #samples, then the Increment and Value List will be automatically updated.)
 - iv. If no Value List is provided and Increment = 0, then the values for the ST are sampled from a uniform distribution across the minimum to maximum interval. If this option is used in an ST that is “Factorial”, the default Increment will be set to 1. Beware that this could create a huge number of ST scenarios, so you should almost always change the Increment to some larger number for a Factorial ST.
6. After an ST is fully specified, as above, hit the “Accept ST scenario” button to save that ST. This saves that ST within your project. Until the ST is accepted, you cannot run that ST. (You can discard an ST that you were building by hitting the Cancel button.) When you save an ST, *VORTEX* saves the specifications for creating all the sampled scenarios in that ST, but does not normally actually save the full scenario information as separate scenarios in the xml Project file. This is because there might be many (maybe 1000s!) of scenarios created by an ST. However, if you do want to save a Project File that has each of the ST sampled scenarios, you can hit the button at the lower right to do so.
7. You can save an ST and exit the ST module (without yet running anything) by hitting the Accept and Close button. In fact, almost always after creating STs, you will want to exit

the ST module to edit the input values (see below) before returning to the STsetup window to run the STs.

8. After Accepting an ST, you can go back to the top dropdown list and select “New ST” to create another ST that defines a set of parameters to be varied (repeating steps #6 and #7 above). Again, be sure to give it a new name (not “New ST”) and be sure to save it by “accepting” after it has been created.
9. Close the STsetup window and go into project Input to edit the STbase scenario that has been created.
 - a. Go to each parameter that you wish to vary in the ST and insert the desired SV into a function for that parameter. (E.g., %FemalesBreeding might be set to “=SV1”)
 - b. Save your *VORTEX* project.
 - c. Return to the STsetup to now run your sensitivity tests.

Running ST scenarios

1. To run the set of scenarios within an ST, first select the saved ST from the top drop-down list, and then re-confirm that all of its settings to be certain that it is the ST you want. If you need to change anything in the ST, remember to Accept those changes before running the tests. Also, after you leave the STsetup, remember to save the overall project again.
2. It can also be useful to run the ST scenarios as population-based models (“Run as Pop-based” button). Population-based models exclude some sources of stochasticity, but they run much more quickly. Thus, they are useful for providing an initial ST that can help you focus later STs on the more important parameters.
3. Hitting the “Run ST” button will run the full set of scenarios created for that ST. When you run the ST sampled scenarios, *VORTEX* will also automatically first run a scenario that uses the base values, and this “base” scenario can later be compared in tables and plots to the sampled scenarios. The graph of population size will be shown during this run of the base scenario, but will not be displayed during all the runs of the sampled scenarios for the ST. You can also run the base scenario back in the main *VORTEX* Run window, and it is sometimes useful to do that before running all the ST scenarios in the ST module – to confirm that the scenario has all required *VORTEX* input; to obtain a baseline set of results; and because this will produce the full set of *VORTEX* output files (including the description of input, deterministic results, etc.) for this baseline case. The results from base scenario run within the ST module and the base scenario run in the main window will be given slightly different labels, so results from both sets of runs are saved (and they may differ due to random stochastic variation).
4. *VORTEX* will normally produce only a subset of the output files for each of the possibly very many scenarios that are created in the ST. These output files summarize the affect of

each tested variable for analysis (see below). If you need a more complete and detailed output from each scenario (e.g., you want the annual census data or summary statistics for each iteration), you can check the option to “Create all population statistic files”. Be aware, however, that depending on which optional output you requested in [Special Options](#), checking this box can lead to a very large number of files created in your project VOutput folder.

5. You can run a set of scenarios for an ST using multiple CPUs, just as you can run multiple scenarios at once in Multi-Vortex [Run](#) mode. There is a risk that conflicts will arise between files being shared (in which case you will get an error message), but usually using 3 or 4 CPUs works as a way to speed up STs with long run-times. If you want to run several different STs at once, you can open several copies of *VORTEX*, open the project, and run the different STs in the separate copies of *VORTEX*. The files accessed by different STs (even in the same project) do not generally overlap, so file conflicts are usually not a problem in this case.

Editing ST scenarios

After creating and saving an ST, you can later return to the STsetup to change some of its settings (but only if you had remembered to accept your ST and then save your *VORTEX* project!). If you change the ST name, your edits will be saved as a new ST and your old one will remain as it was under the old name. If you change the Template, then the old ST and its STbase scenario will be deleted and replaced by the one that you edit and save. If you just change some of the settings for variables to be tested, then the existing ST is edited.

You can remove a variable from an ST by clicking on its row and then hitting the Delete key (the Delete key on your keyboard, not the “Delete ST” button on the STsetup window!). Be aware, however, that when you delete a row from the table, the SV# and GS synonyms for any subsequent variables in the table will be changed.

You can remove an ST (including removing its STbase scenario) by hitting the “Delete ST” button.

Some cautions and hints

1. Be sure to give your ST a new name, replacing the “New ST” label in the top dropdown list.
2. When you are ready to exit the STsetup window, be sure that you have saved your ST (if you have made any changes) by hitting “Accept” or “Accept and Close”.
3. When providing a list of values to be sampled for an ST parameter, separate the values with semi-colons, not with commas.

4. After you first create an ST, don't forget to go into the Input pages to insert the SV1, SV2, etc. into the desired input parameters. Otherwise, the SVs will be varied across the series of scenarios run in the ST, but they will have no effect.
5. When you create and Accept STs, the settings for those STs are saved within your *VORTEX* project. You will then need to save the project before (or when) exiting *VORTEX*.
6. Be careful when you edit the STbase scenario after you create an ST. Any changes that you make in that scenario in the input will apply to all the copied scenarios in the series for that ST. This is, of course, useful in that you can change some parameters after creating the ST, but you need to be aware that if you make edits to the STbase later, they will be applied to the ST.
7. Be aware that the STbase scenario that is created and the ST specifications are intimately linked. If you delete the STbase scenario in the Input tab, then the associated ST specifications will disappear also. If you delete the ST in the STsetup window, then the STbase scenario will be deleted from the scenario list in the project.
8. You can edit the GSvar that is created in the STbase scenario for each SV. This allows you to modify the way that the ST will sample values across the series of iterations run within the ST. For example, you could change the sampling from a uniform distribution across the parameter space to a normal distribution. However, be aware that if you return to the STsetup window and again Accept the ST, *VORTEX* will overwrite any changes that you made to the GS (synonymous SV) variables with the default sampling scheme defined by the STsetup settings.

Output and further statistical analysis of ST results

When ST scenarios are run, *VORTEX* usually produces only a subset of the normal output files, and it uses special extensions for the two files that it does create. If you are running, say, 1000 scenarios within an ST, you normally would not want to see (or have fill up a folder on your disk) 1000 copies of the same or very similar .inp (input), .det (full deterministic results), .yr (detailed census data), .N (a tally of population size for every year of every iteration), and other files. (If you do want to see one set of each of the normal output files, run the base scenario.) The only output files created during the ST runs is a .stdat file for each scenario in the series run for the ST and a .stsum file that contains the final results for each scenario in the series. These files are the same format as the .dat and .sum files created by normal (non-ST) runs of *VORTEX* scenarios.

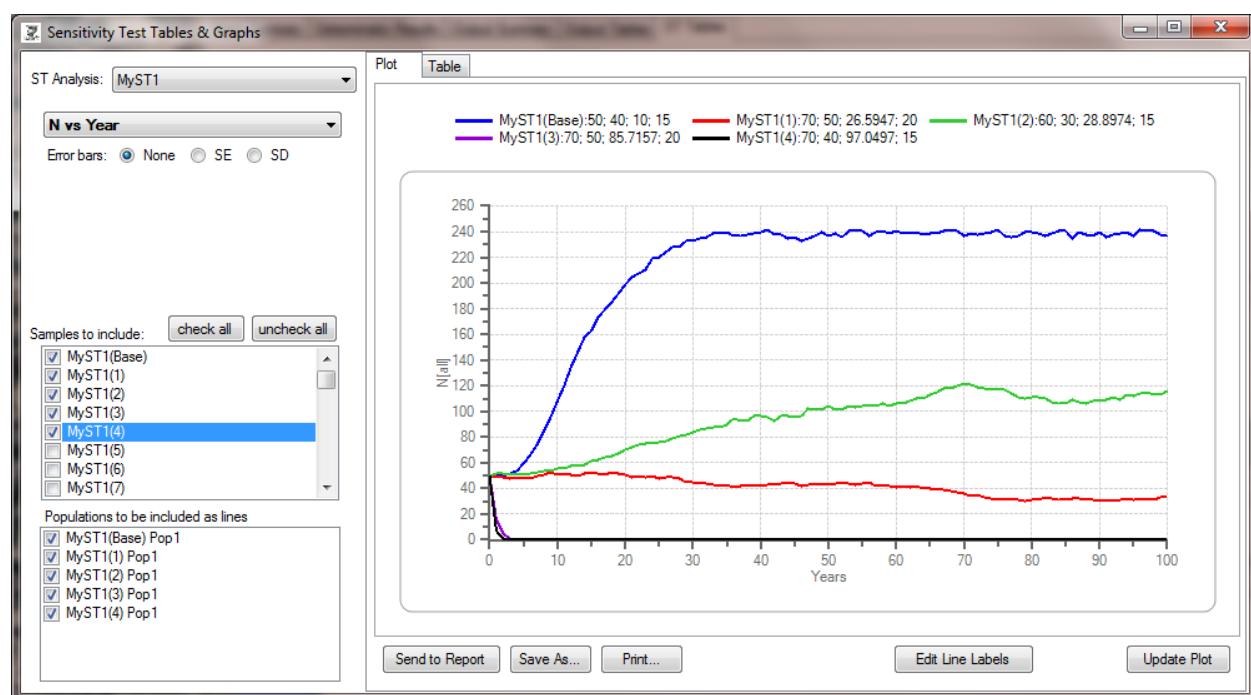
The Text Output contains a tab (ST Tables) to display the summaries of results from Sensitivity Tests that are saved in the .stsum file. The .stsum file can also be opened in Excel, if you specify that columns are delimited with semi-colons. The last few columns of data in the .stsum file are the values of SVs that were sampled in each scenario within the series. This makes it easy to do further statistical analysis (using your favorite statistical package) on the ST results. Commonly, you would want to run a regression analysis or ANOVA of some sort to determine the effect of

each SV (and possible interactions among them) on measures of population viability. For probability of extinction, an appropriate analysis is often a logistic regression. For final population size or population growth rate, often a linear regression is appropriate.

The .stdat files give year-by-year results for each scenario that was run during the ST. Although often you will have no need to examine these files, they do provide the data needed for analyses at intermediate years of the simulation, and they are used for graphing ST results (see next section).

Graphing ST results

Graphs of ST results include the same graphs available for all scenarios, and some additional plots that are useful for comparing the relative effects of variables tested in a ST. All the data that are used for ST graphs are available in files with extension .stdat (with the same data format as the .dat files used for graphing non-ST scenarios) – so the ST results can also be analyzed and graphed in other programs. Note that the Tables & Graphs within ST and the Tables & Graphs tab within the main PMx program are very similar (and that was purposeful), but the ST version will use only the .stdat data files and the main program will use only the .dat data files. Therefore, the data available to be plotted in the two sections are different.

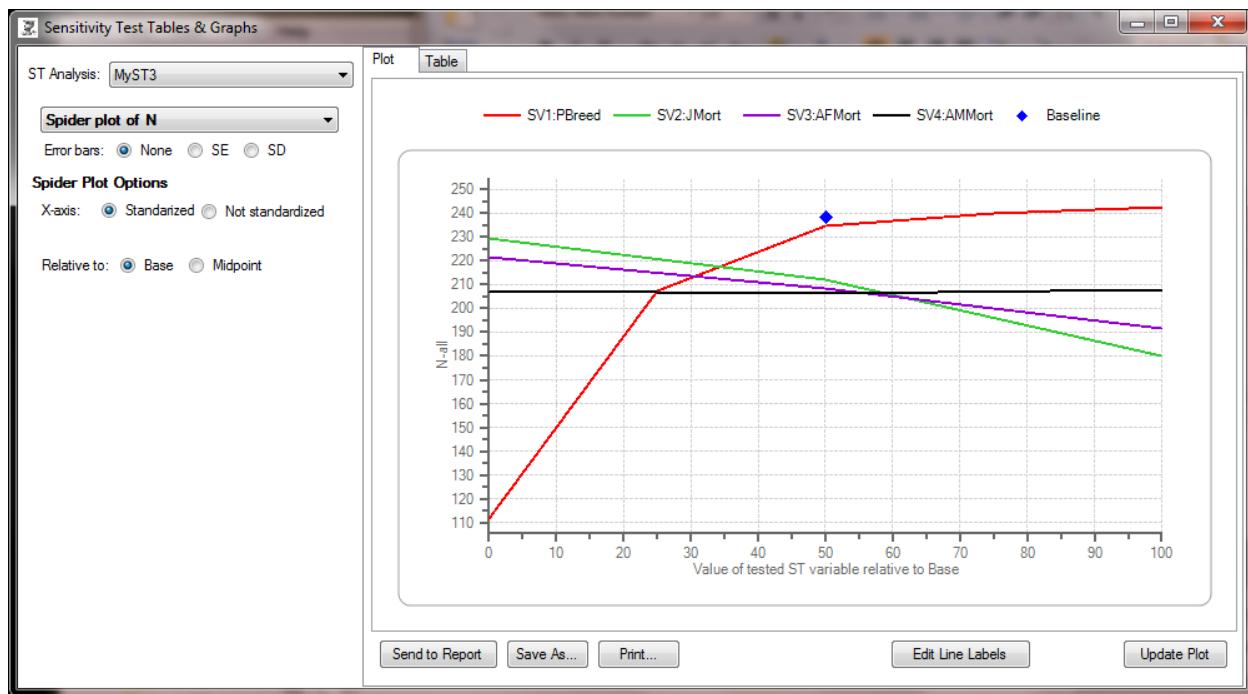


You open the ST Tables & Graphs by clicking on the “Display Graphs” button of the Sensitivity Testing setup window. The first step in generating graphs is to select from the top left drop-down list the ST Analysis for which you want to see the results. You then check which of the sampled scenarios from that ST are to be included in the graph. If your ST generated 100s or more of samples, it might be wise to first select only 5 or 10 to be graphed, until you know that you are getting the results that you want. Also, a graph with 100s or 100s of lines is rarely clear enough to be useful, and may take a long time for the program to create.

After selecting sampled scenarios to be included, the list below it will automatically show the populations for those scenarios. You can leave them all checked, or you could uncheck some. For example, you might want to see only the results for the Metapopulations when your scenarios each have many populations.

From the dropdown below the ST Analysis in the upper left you can select which graph you wish to see displayed. (The Table tab will then also show you the data that generated the graph.) The top three graphs (N, GD, and P[survive]) are the same as the first three default graphs in the main Tables & Graphs tab. Similarly, the Custom Plot option takes you to the settings for specifying any of the standard graphs available in *VORTEX*.

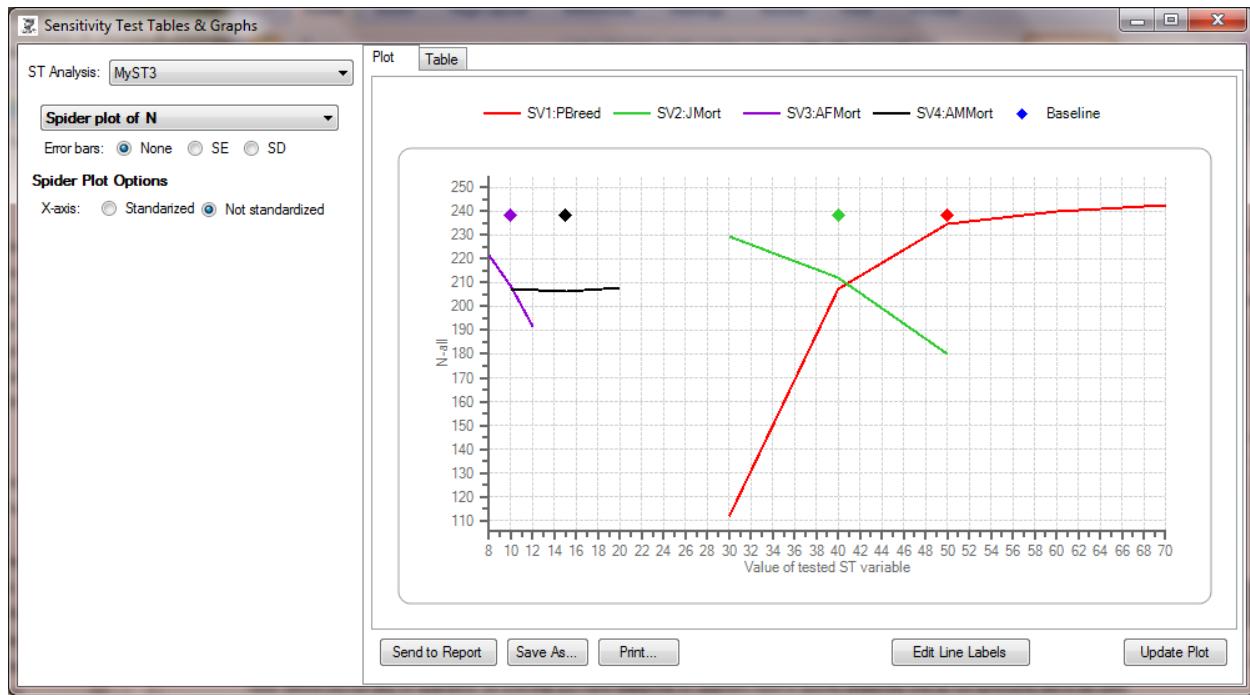
The graphs that are unique to ST are four “spider plots” and four “Max-Min” graphs. These graphs are available only if the ST Analysis used a sampling scheme that generates a discrete set of values for the sampled variable (with an Increment or Value List for every variable to be tested). Also, these specialized graphs always use the full set of samples created by the ST Analysis and only show results for the metapopulations, so you do not select which Sampled Scenarios and Populations to include.



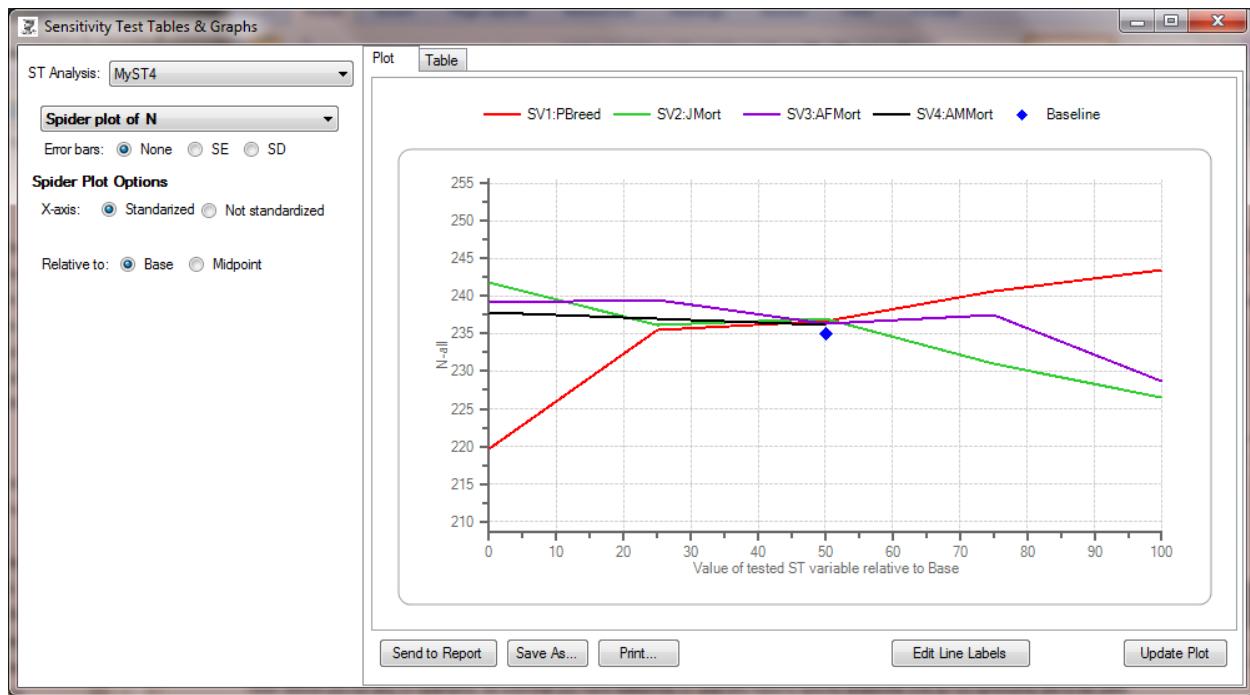
Spider plots show for the discrete values of each variable tested the mean value of the output variable, averaged across all combinations of values for the other tested variables. A steep slope (as for the PBreed variable, at least at the lower range of values, in the above graph) indicates that the test variable has a large impact on the results. A flat or very shallow slope (as for the AMMort variable, adult male mortality, in the above graph) indicates that the variation in that variable had little or no impact on results.

Note that the lines may not be straight, because decreases and increases in a variable may not have symmetrical effects on population dynamics. Also, the lines often will not cross at the value for the base scenario (shown on the graphs as a blue diamond), because of interactions among variables, asymmetrical ranges of values tested, or the “base” values not necessarily being in the middle of the tested ranges of the variables.

There are two options for controlling how the spider plots appear. When the x-axis is standardized, then all variables are plotted on a 0 to 100 scale that covers the range of values sampled. Non-standardized plots (see below) show the lines for each variable plotted against the values that were tested. Because different variables might be on very different scales, often the non-standardized spider plots will have lines that cover discrete segments of the x-axis. (Maybe these are spider plots with the legs pulled off?) In non-standardized plots, the result for the base is plotted at each place along the x-axis that represents the base value of a tested variable.



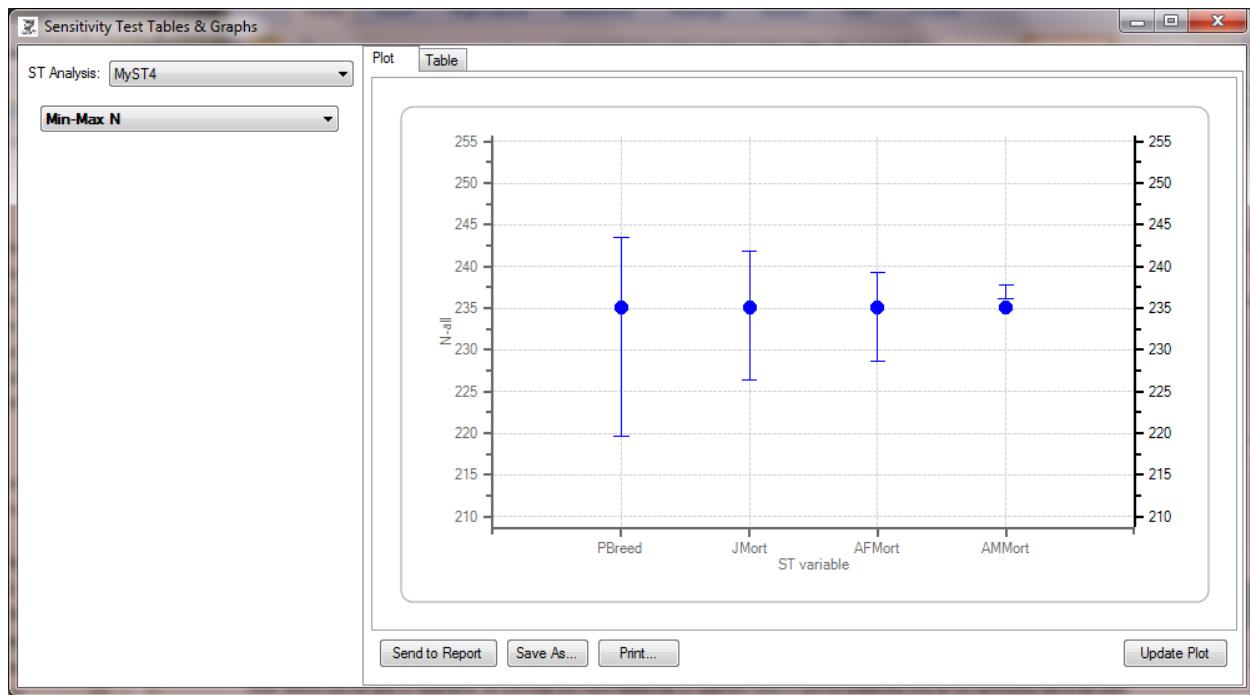
For standardized spider plots, you also have the option of having the x-axis scaled relative to the Base or relative to the Midpoint. If scaled relative to the midpoint, then each line will extend from 0 to 100 on the x-axis scale, with the midpoint of the tested range being plotted in the middle of that scale.



If scaled relative to the Base (see above), then the base value is placed at the $x = 50$ location, and the $x = 0$ or the $x = 100$ point (or both) will be the most extreme tested value(s) for each variable. If the base is not at the midpoint of the tested range for a variable, then the line for that variable will not extend equally far to the right and left of the Base.

The different options for the spider plots give different views of the data. Which view is best depends on your preference and the distributions of values sampled in your Sensitivity Test.

Max-Min plots are a little easier to understand, although they don't provide any information about the range of values that were tested to generate the graphs.



A Max-Min plot shows the range of results for each tested variable, averaged across the values of all other tested variables. A point is placed at the result for the base scenario, but note (as in the above graph) that the ranges tested may not be symmetrical around the base (in terms of the parameter values and/or in terms of model results), and the base may not even have been included within a range that was tested. In a Max-Min plot, the variables with the longest lines had the greatest effect on results.

For all the sensitivity test results, however, it is important to recognize that the results for each variable and even which variables have the biggest impact over their tested ranges depends the ranges of values examined for all the variables (and perhaps even on the other input variables that were not varied in the ST). Due to interactions among factors, adding more variables to an ST can change the picture you get of which variables are the most important drivers of population dynamics. Perhaps the best approach to ST is to include every uncertain variable, tested across its range of uncertainty. This can easily generate a massive number of scenarios in an ST Analysis, but the numbers can be more manageable if you use Latin Hypercube Sampling. Also, after a preliminary ST (maybe with not enough samples to get precise results) show that some variables seem to have very little effect, more thorough sensitivity testing might proceed with that subset of variables that seem most important.

Finally, the graphs available in *VORTEX* for displaying ST results can be quick and effective ways to visualize which uncertain variables are primary determinants of population dynamics, but they are not full replacements for detailed statistical analysis of ST results. For example, complex interactions (possibly involving more than 2 variables) cannot be seen on the graphs, and the graphs do not allow you to select the set of possibly interacting variables that together have largest impact. The results provided in the .stsum file can be fed into statistical packages for regression or other analyses to more fully analyze the determinants of population results.

Using Functions in *VORTEX*

A cautionary note: The use of functions rather than specific values for input into Vortex is both very powerful and often difficult to use wisely. For many users, there will never be a reason to use functions in the input. For those who do need greater flexibility to build more complex models, Vortex allows you to set most of the input variables (almost all of the ones that would normally accept a number) to be functions of properties of the system, the populations, or individuals.

VORTEX provides the option of modeling demographic rates as functions of population or individual parameters. The population descriptors that can be used as variables in the functions include time (year in the simulation), iteration, population, population size, carrying capacity, numbers of juveniles (animals in the first age class), subadults (greater than 1 year, but not yet breeding age), adult females, adult males, all females, or all males, and gene diversity (expected heterozygosity). Individual characteristics that can be entered as variables in these functions include ID#, sex, age, number of mates (0 or 1 for females and monogamous males; possibly more for polygamous males), inbreeding coefficient, and genotypes at modeled loci. Almost all demographic rate parameters – such as the percent of females breeding each year, environmental variation in breeding, litter/clutch size, sex ratio, mortality rates, environmental variation in mortality, catastrophe frequency and severities, carrying capacity, dispersal, dispersal mortality, occurrence of harvest and supplementation, and definition of extinction – can be specified to be functions of the above population and individual variables. Functions are evaluated each time that the demographic rate is called (e.g., each year for a function for carrying capacity or numbers of individuals harvested or supplemented; for each individual in each year for a function of a reproductive parameter, mortality, or dispersal rate).

Functions return rational numbers (positive or negative numbers that may include digits after a decimal delimiter), calculated to double-precision (8-byte numbers, with about 17 significant digits). However, for speed, within the program many rates will be evaluated with fewer significant digits (for example, inbreeding coefficients are calculated to 5 significant digits, mortality rates greater than 0.999999 are rounded to 1.0, and dispersal rates smaller than 0.00001 are rounded to 0.0.). If the use of the value requires an integer (e.g., a number of individuals to be harvested), *VORTEX* may truncate, round, or probabilistically round the value to obtain an integer (see descriptions of input rates, above), so it is best to include the intended type of rounding within the function itself so that the user controls exactly how non-integer values will be treated.

The flexibility to specify population rates as functions rather than as fixed constants allows users to model specific population dynamics that might be known to be appropriate for some species, or that are of interest in a theoretical analysis. With some creativity and perhaps considerable effort, *VORTEX* can now model many of the kinds of population dynamics that can be envisioned. As just a few examples:

- it might be known that carrying capacity will change at some determined date in the future;

- it might be believed that reproductive rates will change over time, perhaps due to some management action;
- the density dependence observed in reproduction might not fit the shapes of the curves allowed in previous versions of *VORTEX*;
- mortality rates might change over time, or respond in a complex way to population density;
- inbreeding might impact fecundity, adult survival, or might affect the two sexes differently;
- dispersal might be age and sex dependent;
- fecundity, mortality, or the effects of catastrophes might be age-dependent;
- environmental variation might occur with a periodicity that is longer than a year, or catastrophes might have multi-year effects.

Note that *VORTEX* includes within the input the option to model reproduction as a density dependent function, and an option to model carrying capacity as having a linear change over a specified number of years. Easy access to these two particular functions are provided because they are needed more frequently than are detailed functional dependencies of most other rates. Even for these two rates, however, you can specify these functions to have almost any shape if you use the function editor to specify the rates.

For most users and for most purposes there will be no need to model demographic rates as functions; it is usually fully adequate to specify fixed demographic rates rather than functions. Specification of rates as functions can be difficult: the appropriate form of the function is rarely known, the function parameters are usually very difficult to estimate, and it is not trivial to enter a function correctly. If alternative functions need to be examined in sensitivity testing, the number of combinations of input parameters to be explored can quickly become overwhelming. Consequently, we would not recommend that novice users or students use the function option within *VORTEX*.

Primary changes in Functions from *VORTEX 9* to *VORTEX 10*

This section summarizes important aspects of functions that have changed in *VORTEX 10*.

Considerably more pre-defined variables are now available for use in functions, some of the old (version 9) variables have changed or been dropped, and the syntax of functions has changed a little. Therefore, it is important to understand the changes if you will be opening old version 9 projects containing functions.

The variable for inbreeding (*I*) is expressed as a proportion (0 to 1 scale), rather than a percent.

Default values for any variables used in functions can be specified in the input on the State Variables page. If specified, these values will be used when evaluating functions at the start of a simulation for determination of deterministic growth rates and stable age distribution. For any variables pertaining to individuals (e.g., age, sex, genotypes), these default values will be used each year when evaluating functions that are needed for population-level calculations, such as

the number of pairings to reach K (if that option is chosen) or functions defining harvest or supplementation.

Order of precedence of operators, within any parentheses, follows standard rules, rather than the strict left to right application of operators of *VORTEX* 9. Order of precedence of operators (evaluated first at the top), with left-to-right evaluation for sets of operators on the same line, is:

^ (exponentiation)
- (negation)
! (logical NOT)
*, /, % (multiplication, divide, and modulus [remainder from integer division])
+, -
<, >, <=, >=
=, != (“=”,”==”, and “EQUALS” are all equivalent; and “#” is the same as “!=”) AND, OR, NAND, NOR (“&” and “&&” and “AND” are all equivalent; “|” and “||” and “OR” are all equivalent; “NAND” can be coded as “\$” and “NOR” can be coded as “~”.)

Syntax for using GSvars, PSvars, ISvars, and related functions has changed, as follows:

GS1 [rather than GS(1)]
PS1 [rather than PS(1)]
PPS1(p) [rather than PPS(1;p)]
IS1 [rather than IS(1)]
ITOT1 [rather than ITOT(1)]
ISUM1 [rather than ISUM(1)]
IMEAN1 [rather than IMEAN(1)]
IMAX1 [rather than IMAX(1)]
IMIN1 [rather than IMIN(1)]

No longer available as built-in variables (but available via PSvars set to these values): FF, MM, JJ, UU, WW, XX.

NN is still available, but now with syntax NNx (e.g., NN1, rather than NN(1))

Parentheses – In *VORTEX* 9, the number of closing parentheses at the end of a function were automatically adjusted to make them match the opening parentheses. However, this is a dangerous thing for *VORTEX* to do, because the error in parentheses may not actually have been at the end of the function. Therefore, in Vortex10, an error will occur if parentheses do not match in a function.

A number of new built-in variables and functions are available for use. See the list below for the set of variables, operators, and functions now available.

Specification of Demographic Rates as Functions

Dependencies of demographic rates on population and individual parameters are entered into *VORTEX* by specifying the functional relationships. There are two ways that you can enter a function rather than a constant for an input variable: you can type the function directly into the input box for specifying the rate, or you can open a Function Editor to help you develop the function to describe the relationship. If you type a function directly into an input box, you must precede the function with an “=” sign (to distinguish the specification of a rate as a function rather than as a constant). When you finish editing a function in the Function Editor, the Function Editor can insert the function back into the active input box when you accept the function. It is usually easier (and safer) to build a function first within the Function Editor and then send it over to the input page.

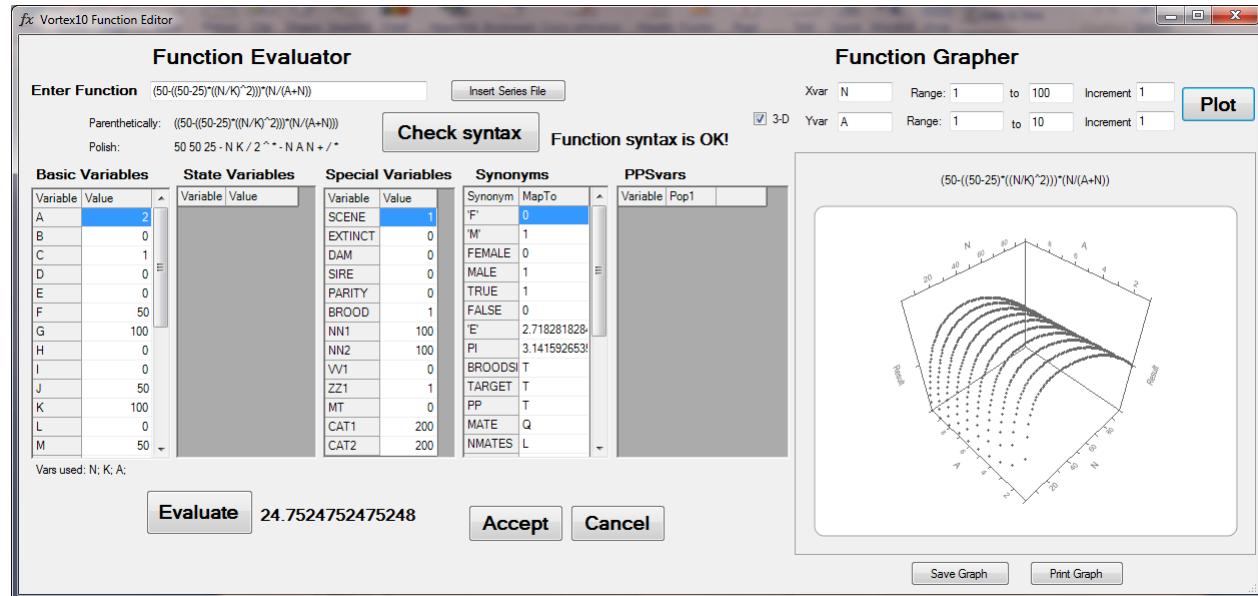
(Parentheses), [brackets], and {braces} may be used interchangeably to indicate the order of operations.

The case of function names and variables is ignored. All letters that are entered in a function within the Function Editor are converted to upper case by *VORTEX*. The separator used between variables in a binary operator must be a semi-colon (;) – not a comma (,) – to avoid problems arising from the comma being used as a decimal separator with some regional data settings.

Hints for speed: Functions can be much slower to evaluate than are simple constant values. Therefore, don't enter **=0.25** or **=1/4** (both interpreted as functions, because they are preceded by ‘=’) when you could have entered just **0.25**. And don't use the same complicated function in multiple places. Instead, set a PSvar to that function and then set the various variables to that PSvar. This also makes it easier to later edit that function. Random numbers (RAND, NRAND, and especially SRAND and SNRAND) are very slow. Therefore, don't use them more than needed. For example, instead of using **=SRAND(Y+123)** three times to get the same random number used in all 3 places (which will require 3 calls to the SRAND function each year), create a PSvar that **=RAND**, and then use this PSvar in all three places (so that RAND gets called once per year).

Using the Function Editor

The Function Editor in *VORTEX* provides a tool for editing, testing, and graphing functions that you might want to use in your Project. The Function Editor can be accessed either by hitting the function icon (*fx*) or with **ctrl-F**.



If you are on a data entry box that will accept a function when you enter the Function Evaluator, the function (perhaps just a number) from that input data will be transferred to the top left “Enter Function” box. If you are not on a valid data entry box for a function, then the Function Editor will start with a simple default function (**A%3+RAND**). In either case, you can type or edit any function into the top left box. The “Insert Series File” button allows you to browse to find a file that you want to insert into a “FILE(filename)” within your function. In the Function Editor, it is optional to precede your function with an ‘=’ sign.

The “Check Syntax” button can be used to test if the function is considered valid by *VORTEX* (all variables recognizable, parentheses matched, and required number of values passed to built-in functions and operators). This check will also display the Function with extra parentheses added to make clear the order of operators, and will give the function in Reverse Polish Notation to further verify that the function and its order of operators is being interpreted correctly by *VORTEX*.

You can (and should!) evaluate the function with any specific set of variable values in order to see if it is returning the result that you expect. First, in the five tables of variables (some of which may be empty if your scenario has no State Variables) enter the value that you want assigned to any variable that is used in your function. A handy list of “Vars used” is provided at the bottom of the tables. (However, sometimes that list can be confusing, because *VORTEX* re-uses some synonyms for variables that would never be used in the same step in the simulation. For example,

BROODSIZE (only used during Reproduction) is coded as **T**, as is the **TARGET** population (used in Dispersal). When *VORTEX* shows the “Vars used” it sometimes displays the other variable that has the same 1-letter code, but it doesn’t matter, because the synonyms are interchangeable in functions.) Hit the Evaluate button to see what value your function returns with those values for the variables. If the Evaluate button is disabled, that means that your function had invalid syntax and needs to be edited. The Plot button will be disabled if the function syntax is invalid.

As with other graphs in *VORTEX*, you can print or save the graph, and you can open the graph Properties by double-clicking on the graph.

After you finish editing and testing a function, you can hit Cancel or the close window icon (‘x’) to return to the *VORTEX* data entry pages. If you hit “Accept”, then the function will be transferred into the current input data box, if it was a box that allows a function to be entered. (Otherwise, the “Accept” button will not be accessible.)

Variables and Operators Available for Use in Functions

The following tables list the many variables and operators that are available for use in *VORTEX* functions. It is important to be aware when the variables are updated as the simulation is running. For many variables, it is obvious; for example, the variable **Y** (for year) is incremented at the beginning of each year of the simulation. But population tallies, such as **N**, **F**, **M**, **J**, and **G**, are updated before each event (e.g., Breed, Mortality, Harvest) in the annual cycle. Thus, for example, **N** at the beginning of Mortality will be the number of individuals after the reproduction event has occurred (unless you changed the sequence of events in the annual cycle). Tallies of reproduction (**PAIRS**, **BROODS**, **PROGENY**) are incremented as each individual reproduces, and this permits you to modify the breeding rates of subsequent individuals based on the reproductive success already accumulated. Dispersal tallies (**IMMIGRANTS** and **EMIGRANTS**) are updated within the dispersal process. However, you should not use these two variables to modify dispersal rates themselves, because it is not predictable in what order individuals will disperse or when the dispersal tallies are updated within the dispersal event. Harvest and Supplementation tallies (**HARVESTS** and **SUPPLEMENTS**) are updated before each population is subjected to each annual event. Therefore, it is possible to modify the harvest or supplementation of 2nd and later populations based the harvests and supplementations already imposed on prior populations that year. Finally, note that the various tallies of activity within each year (**PAIRS**, **BROODS**, **PROGENY**, **IMMIGRANTS**, **EMIGRANTS**, **HARVESTS**, and **SUPPLEMENTS**) are re-set at the beginning of each year. Thus, if you want to use the tally from the prior year, you must set a PSvar to the value at the end of the year, and then use that PSvar in the next annual cycle.

Valid Function Variables: Scenario descriptors

| | | |
|----------|---|----------------------------|
| R | = | Run (simulation iteration) |
| Y | = | Year |

ITOT1, ITOT2, etc. = Total, across living individuals, of **i**th individual state variable.

IMEAN1, IMEAN2, etc. = Mean, across individuals, of **i**th individual state variable.

IMIN1, IMIN2, etc. = Minimum, across individuals, of **i**th individual state variable.

IMAX1, IMAX2, etc. = Maximum, across individuals, of **i**th individual state variable.

Note: when the above four kinds of variables are used in definitions of Global State Variables, the calculations are done over the metapopulation. Otherwise, they apply to the specific population. Caution: if you use any of the above within the definition of an Individual State Variable, results can be unpredictable because you cannot be certain when the tallies are updated while ISvars are themselves being changed. (If there is just one population, the values used during an ISUpdate will be those calculated before the update was started.)

Miscellaneous other variables

| | |
|---------------|---|
| ST | True/False code for whether the scenario is an ST sample |
| SCENE | number of the scenario being run |
| CENSUS | number of the current census; same as Y if there is one census event per year. |

Valid Function Variables: Population descriptors

| | | |
|---|---|--|
| D | = | Dispersal rate (from the matrix entered), used only in functions modifying dispersal. |
| F | = | Number adult females |
| G | = | Percentage of initial gene diversity (expected heterozygosity) in the population |
| I | = | Mean inbreeding for a population, calculated as % homozygosity |
| J | = | Number of juveniles (age 0-1) |
| K | = | Carrying capacity |
| M | = | Number of adult males in the population |
| N | = | Population size |
| P | = | Population identifier |
| T | = | recipient (target) population for dispersal or translocation (synonym PP) |
| U | = | Number of subadults (age > 1, < breeding age) |
| X | = | Number of females (all ages) |
| W | = | Number of males (all ages) |
| NNP | = | N for population p . |
| PP | = | Recipient population of a dispersing individual, for use in Dispersal Modifier functions. |
| PS1, PS2, PS3, ..., PS9 | = | Population State Variables previously defined. |
| PPS1(p), PPS2(p), etc. | = | Population State Variables for population p . |
| CAT(c) | = | Number of years since last occurrence of Catastrophe type c . (CAT(c) > maximum years, if catastrophe has not yet occurred.) |
| GDIST(1;2) | = | Genetic distance between population 1 and 2 . |
| GDIST(1;1) | = | Genetic distance between population 1 and the metapopulation. |
| The metric used can be specified in the Special Options . The default is Nei's standard genetic distance. | | |
| ITOT1, ITOT2, etc. | = | Total, across living individuals, of i th individual state variable. |
| (ISUM1, ISUM2, etc.) | = | Synonyms for ITOTi .) |
| IMEAN1, IMEAN2, etc. | = | Mean, across individuals, of i th individual state variable. |
| IMIN1, IMIN2, etc. | = | Minimum, across individuals, of i th individual state variable. |
| IMAX1, IMAX2, etc. | = | Maximum, across individuals, of i th individual state variable. |
| Note: when the above four kinds of variables are used in definitions of Global State Variables, the calculations are done over the metapopulation. Otherwise, they apply to the population. | | |
| PAIRS | | number of pairs that have been created in that year. PAIRS will be updated after each pair has been created. This allows you to control the probability of future pairs based on the number already made. |
| BROODS | # | broods produced within that year; updated as broods are made. |
| PROGENY | # | progeny produced within that year; updated as progeny are made. |
| BREEDRECS | # | pairs last calculated to be needed, when "Breed to K" option is chosen. |
| EMIGRANTS | # | individuals dispersing from the population in the current year; updated as dispersal occurs, but do <i>not</i> use in a function within the Dispersal event, as it is unpredictable when the value changes during dispersal. |
| IMMIGRANTS | # | individuals dispersing into the population in that year. |
| HARVESTS | # | individuals harvested in that year; updated after each population is harvested. |
| SUPPLEMENTS | # | individuals supplemented into the population that year; updated after each population is supplemented. |
| TRUNCATED | # | individuals removed that year due to carrying capacity truncation. |
| EXTINCT | | True/False code for whether the population is currently extinct. |

Valid Function Variables: Individual descriptors

| | | |
|----------|---|---|
| A | = | Age |
| I | = | Inbreeding coefficient (a number from 0 to 1) |
| P | = | Population |
| S | = | Sex (0 or 'F' for female, 1 or 'M' for male) |
| V | = | Paternal allele identifier |
| Z | = | Maternal allele identifier |

Note that **I** has a different meaning when applied to individuals (the individual's inbreeding) vs populations (population mean homozygosity). The meaning will depend on where the function is used.

VV(i) and ZZ(i) = the paternal and maternal alleles at the **i**th modeled locus.

Can also be written as **VV1, VV2, ... and ZZ1, ZZ2**, etc.

IS1, IS2, IS3, ... , IS10 = Individual State Variables.

OIS1, OIS2, OIS3, etc. = ISvars of another individual (specified by an OID state variable)

DAM1, DAM2, DAM3, etc. = ISvars of the Dam (*during the breeding cycle, these are set to the ISvars of the dam in the pairing, not the dam of the female in the pairing*)

SIRE1, SIRE2, SIRE3, etc. = ISvars of the Sire (*during the breeding cycle, these are set to the ISvars of the mate or potential mate, not the sire of the female in the pairing*)

X1, X2, X3, etc. = Individual State Variables of an animal when setting initial kinships.

Y1, Y2, Y3, etc. = Individual State Variables of the other animal when setting initial kinships.

PX = Population of an animal when setting initial kinships.

PY = Population of the other animal when setting initial kinships.

ID numerical code for an individual, set sequentially as individuals are created.

INDEX 1-based index of an individual (number in the sequence of current individuals).

0 if dead. (Can be used, for example, to test if a DAM is alive.)

MT mtDNA haplotype

MK mean kinship

DAMID ID of dam; can also be written as **DAM**

SIREID ID of sire; can also be written as **SIRE**

KIN kinship between parents

LOCUS locus (for mutation models), -1 for mtDNA

MATE ID of mate, ID of first mate for polygynous males, -1 when not paired

NMATES # mates of a male

PAIRTENURE # years a female has been paired to her mate; 0 in the year when paired;
-1 when not paired

MALLELE 1 for maternal allele, 0 for paternal allele, used only for determining mutation rate

PARITY # offspring produced in an individual's lifetime (to that point in time)

BROOD brood number for the individual's mother in that year; i.e., 1 if from a first brood

BROODSIZE # offspring born within current brood

DIST Euclidean distance between potential mates (only if X and Y exist as ISvars)

MK1, MK2, MK3, ... Mean kinship of the individual to population 1, 2, 3,

MKMETA Mean kinship of the individual to the metapopulation.

KNOWN The proportion of an individual's genome descended from known founders, used when Unknown founders in a studbook are excluded from calculations (see Special Options).

Note that the following variables are available only during the breeding cycle, as pairs are created and offspring produced: **DIST, KIN, BROOD, and BROODSIZE**.

The variables for accessing the Individual State variables of a dam (**DAM1**, **DAM2**, etc.), sire (**SIRE1**, **SIRE2**, etc.), or other individual (**OIS1**, **OIS2**, etc.) can be accessed by functions at any time during the annual cycle, rather than just when an individual is being created (as was the case in versions prior to 10.3). If the dam, sire, or other individual does not exist (e.g., an individual is a founder and, therefore, it does not have a known dam or sire) or is dead, the values of its state variables will be 0. Thus, one method to determine if, for example, an individual's dam has died is to check if one of its state variables = 0. When the simulation is within the breeding cycle (i.e., mates are being selected, brood sizes are being set, and offspring being created), the DAMx and SIREx variables refer to the parents (or potential parents) of the brood being created, and not to the dam and sire of the female that is producing that brood.

Valid Vortex Operators

| Function | Description | Example |
|------------------|-----------------------|--|
| ABS | Absolute value | ABS(-10) = 10 |
| NEG | Negative | NEG(-10) = 10 |
| CEIL | Ceiling | CEIL(3.12) = 4 |
| FLOOR | Truncate | FLOOR(3.12) = 3 |
| ROUND | Round | ROUND(3.12) = 3 ; ROUND(5.5) = 6 |
| PROUND | Probabilistic Round | PROUND(3.12) = 3 (88% of the time) or 4 (12% of the time) |
| SQRT, SQR | Square root | SQR(1.44) = 1.2 |
| LN, LOG | Natural logarithm | LN(1.60) = 0.47 |
| LOG10 | Base 10 logarithm | LOG10(1.60) = 0.20412 |
| EXP | e raised to the power | EXP(0.47) = 1.60 |
| + | Addition | 1.0+2.0 = 3.0 |
| - | Subtraction | 2.0-1.0 = 1.0 |
| * | Multiplication | 2.0*3.0 = 6.0 |
| / | Division | 6.0/2.0 = 3.0 |
| POW, ^ | Exponentiation | POW(10;0.20412) = 10^0.20412 = 1.60 |
| MAX | Maximum | MAX(3.12;4.21) = 4.21 |
| MIN | Minimum | MIN(3.12;4.21) = 3.12 |
| MOD, % | Modulus | MOD(33;8) = 33%8 = 1; MOD(33.5;5) = 3.5 |

Defined Constants

| | | |
|------------|---------------|------------------------------|
| PI | 3.1415927 ... | SIN(PI/4) = 0.7071067 |
| 'E' | 2.7182818 ... | LN('E') = 1.0 |

| Function | Description | Example |
|------------------------------------|----------------------------|---|
| Trigonometric Operators | | |
| SIN | Sine | SIN(PI/2) = 1.0 |
| COS | Cosine | COS(PI/2) = 0.0 |
| TAN | Tangent | TAN(PI/4) = 1.0 |
| ASIN | Arcsine | ASIN(1.0) = 1.5707963 |
| ACOS | Arccosine | ACOS(0.0) = 1.5707963 |
| ATAN | Arctangent | ATAN(1.0) = 0.7853981 |
| DEGREES | Convert Radians to Degrees | DEGREES(PI/4) = 45 |
| RADIANS | Convert Degrees to Radians | RADIANS(45) = PI/4 = 0.7854 |
| Logical (Boolean) Operators | | |
| ==, = | Is equal to | (3=2) = 0 = FALSE |
| NOT, ! | Negation | !(3=4) = 1 = TRUE |
| !=, # | Not equal to | (3!=4) = 1 |
| AND, && | And | ((3=4)AND(3!=4)) = 0 |
| OR, | Or | ((3=4)OR(3!=4)) = 1 |
| > | Greater than | (3>4) = 0 |
| < | Less than | (3<4) = 1 |
| >= | Greater than or equal to | (3>=3) = 1 |
| <= | Less than or equal to | (3<=3) = 1 |
| COMPARE | Compare two values | COMPARE(A;B) = 1 (if A>B), or -1 (if A<B), or 0 (if A=B) |
| IF | Conditional evaluation | IF(A>B;5;10) = 5 (if A>B) or 10 (if A<=B) |
| TRUE | 1.0 | (10>5) = TRUE |
| FALSE | 0.0 | !TRUE = FALSE |

| Function | Description | Example |
|---------------------------------------|--|--|
| <h2>Sampling Distributions</h2> | | |
| RAND | Uniform random (0 – 1) | RAND = 0.2341 or 0.8714 or ... |
| NRAND | Normal random deviate | NRAND = 0.512 or -0.716 or ... |
| UNIFORM | Uniform random (A – B) | UNIFORM(1;5) = 1.5 or .3.6 or ... |
| IUNIFORM | Random integer(A – B) | IUNIFORM(1;5) = 1.0 or 2.0 or ... |
| POISSON | Poisson, with mean m | POISSON(3) = 0.0 or 1.0 or 2.0 or ... |
| POISSON1 | 0-truncated Poisson; sampled from Poisson | POISSON1(3) = 1.0 or 2.0 or ... |
| SRAND | A “seeded” random number; hence, SRAND(x) provides a random number between 0 and 1 with the seed value x | |
| SNRAND | A “seeded” random normal deviate; hence, SNRAND(x) returns a number from a (0,1) normal distribution with the seed value x | |
| SUNIFORM | A “seeded” random uniform; hence, SUNIFORM(a;b;x) provides a random number between a and b (inclusive) with the seed value x | |
| SIUNIFORM or SIRAND | A “seeded” uniform random integer; hence, SIUNIFORM(a;b;x) provides a random number between a and b (inclusive) with the seed value x | |
| SPOISSON | A “seeded” random number generator; hence, SPOISSON(m;x) provides a random number from a Poisson distribution with mean m with seed value x | |
| SPOISSON1 | A “seeded” 0-truncated Poisson; SPOISSON1(m;x) provides a random number greater than 0 from a Poisson distribution with mean m with seed value x | |
| BINOMIALN SBINOMIALN | The number of successes sampled from a binomial distribution with parameters mean m and sample size n.; SBINOMIALN(m;n) uses seed value x. | |
| BINOMIALP SBINOMIALP | The proportion of successes sampled from a binomial distribution with parameters mean m and sample size n.; SBINOMIALP(m;n) uses seed value x. | |
| BETA, SBETA | A beta distributed random number; BETA(a;b) samples a random number from a beta with shape parameters a and b. SBETA(a;b;x) uses seed value x | |
| BETAM, SBETAM | A beta distributed random number; BETAM(m;s) samples a random number from a beta with mean m and SD s. SBETAM(m;s;x) uses seed value x | |
| GAMMA, SGAMMA | A gamma distributed random number; GAMMA(a;b) samples a random number from a gamma with shape parameters a and b. SGAMMA(a;b;x) uses seed value x | |
| GAMMAM, SGAMMAM | A beta distributed random number; GAMMAM(m;s) samples a random number from a gamma with mean m and SD s. SGAMMAM(m;s;x) uses seed value x | |
| DECAY, SDECAY | An integer sampled from the distribution representing exponential decay at a constant rate; DECAY(p) samples a number of timesteps until transition in a process with probability p each timestep. SDECAY(p;x) uses seed value x. If p<=0, then the function returns 0 (not infinity). | |
| DISCRETE, SDISCRETE | An integer sampled from a discrete distribution with relative probabilities specified by the parameter list. DISCRETE(a;b;c;...) returns 1, 2, 3, ... with relative probabilities of a,b,c,... SDISCRETE(x;a;b;c;...) uses seed value x. | |

The operators to create, modify, and operate on lists will not be used by most *VORTEX* users, but they can be powerful ways to create specific sequences of values. These can then be used to set a different value of a demographic rate for each age class or at each year or in each population. Some of these functions are used by the automated Sensitivity Testing module to sample from the lists of values to be tested.

| Function | Description | Example |
|----------------------------|---|--|
| Operations on Lists | | |
| LOW | Lowest value | LOW(3;5;7;1;8) = 1.0 |
| HIGH | Highest value | HIGH(3;5;7;1;8) = 8.0 |
| MEDIAN | Median value | MEDIAN(3;5;7;1;8) = 5.0 |
| MEAN | Mean value | MEAN(3;5;7;1;8) = 4.8 |
| SUM | Sum | SUM(3;5;7;1;8) = 24.0 |
| PRODUCT | Product | PRODUCT(3;5;7;1;8) = 960.0 |
| LOOKUP | Value at the X location in the subsequent list | LOOKUP(3;5;7;1;8) = 1.0 |
| CHOOSE | Randomly chosen value from the list | CHOOSE(3;5;7;1;8) = 3 or 5 or 7 or 1 or 8 |
| SCHOOSE | Seeded randomly chosen value from the list | SCHOOSE(x;3;5;7;1;8) = 3 or 5 or 7 or 1 or 8. with seed x |
| COUNT | Number of values | COUNT(3;5;7;1;8) = 5.0 |
| FIND | Location of value X in the subsequent list (0 if none) | FIND(1;5;7;1;8) = 3.0 |
| PERCENTILE | Value in the subsequent list that is at the X percentile | PERCENTILE(40;5;7;1;8) = 7.0 |
| PRANK | Percentile of value X in the subsequent list | PRANK(3;5;7;1;8) = 5.0 |
| TIMESERIES | List item at the location given by the current year, Y | TIMESERIES(3;5;7;1;8) = 5.0 in year 2 |

Note that the following operations to create or modify lists can't stand alone in a function, because they do not return a single value. Instead, they must be used within one of the above functions that operate on a list. For example, a function of “=SORT(1;3;5)” is not valid, but “=FIND(3;SORT(1;3;5;4;2))” is valid and returns the answer 3.

| Function | Description | Example |
|---|--|---|
| Operations to create or modify Lists | | |
| FILE | Creates a list from the values in a text file | FILE("C:/Temp/MyFile.txt") |
| FILECOL | Creates a list from the values in a column or row of a file | FILECOL(C:/Temp/MyFile.txt;2) |
| FILEROW | Gets the value at row r and column C in a text file | FILEROW(C:/Temp/MyFile.txt;3) |
| FILECELL | | FILECELL(MyFile.txt;2;3) |
| PLIST | Creates a list from the values in PPSx | PLIST(1) = PLIST1 = (PPS1(1);PPS1(2);PPS1(3); etc.) |
| I LIST | Creates a list of values of ISx in living individuals | I LIST(1) = I LIST1 = (IS1(ind 1); IS1(ind 2); IS1(ind 3); etc.) |
| SEQUENCE | A list with values from a to b , by increments of c | SEQUENCE(1;6;2) = (1;3;5) |
| SORT | Sorts from low to high | SORT(1;7;3;5;8) = (1;3;5;7;8) |
| SORTREV | Sorts from high to low | SORTREV(1;7;3;5;8) = (8;7;5;3;1) |
| REVERSE | Reverses the order | REVERSE(1;7;3;5;8) = (8;5;3;7;1) |
| SUBSET | creates a list from c; d; ... , starting with the a position and ending at b position | SUBSET(2;4;6;7;5;8;9) = (7;5;8) |
| SAMPLE | Creates a list of x items randomly chosen. | SAMPLE(3;1;7;3;5;8) = (1;5;7) or (8;1;3) or (3;1;5) or ... |
| SSAMPLE | Creates a list of x items chosen with seed s | SSAMPLE(123;3;1;7;3;5;8) = (1;5;7) or ... with the same list returned every time. |
| APPEND | Appends x to the list | APPEND(2;1;7;3;5) = (1;7;3;5;2) |
| PREPEND | Inserts x at the beginning | PREPEND(2;1;7;3;5) = (2;1;7;3;5) |
| INSERT | Inserts x at the n th position | INSERT(6;2;1;7;3;5) = (1;6;7;3;5) |
| REMOVE | Removes first item equal to x | REMOVE(3;7;3;5;3;8) = (7;5;3;8) |
| REMOVEALL | Removes all items equal to x | REMOVEALL(3;7;3;5;3;8) = (7;5;8) |
| REMOVEAT | Removes the n th item | REMOVEAT(3;7;3;5;8) = (7;3;8) |
| State Variable Histories | | |
| GSHIST | Gets a value from the history of the GSvar | GSHIST1(3) = 3rd value assigned to GS1 by its Transition fn. GSHIST1(0) = Init value. |
| PSHIST | Gets a value from the history of the PSvar | PSHIST3(2) = 2nd value assigned to PS3. PSHIST3(0) = Initialization value. |
| ISHIST | Gets a value from the history of the ISvar | ISHIST2(5) = 5th value assigned to IS2. ISHIST2(0) = Initialization value. |
| GSLIST | List of GSvar history. | GSLIST1 = GSHIST1(0);GSHIST2(1);... |
| PSLIST | List of PSvar history. | PSLIST3 = PSHIST3(0);PSHIST3(1);... |
| ISLIST | List of ISvar history. | ISLIST2 = ISHIST1(0);ISHIST2(1);... |

FILE(filename) creates a list from the values in a text file. E.g., **LOOKUP(3;FILE(myfile.txt))** will look up the third value in the list provided in myfile.txt. This can be useful if you want to specify a series of values in a file that will be read into *VORTEX*. The filename should normally contain the complete path, but *VORTEX* will try to find the file in the project directory (where the project xml file is located) if no directory is specified. The filename can be in quotes or not, but it is safer to use the quotes, because then any punctuation in the filename will not be misinterpreted as a delimiter between parameters. The filename cannot contain parentheses.

For the next three operators, the rows and columns must be specified as constant numeric values, and cannot be variables (such as **A** or **N**), because the file is read once at the beginning of the scenario simulation to obtain the row, column, or cell values.

FILECOL(filename;n) creates a list from the values in the nth column of a text file (with columns of each row separated by semi-colons, tabs, or spaces, but not commas). E.g., **LOOKUP(3;FILECOL(myfile.txt;4))** will look up the third value in the 4th column of data in myfile.txt.

FILEROW(filename;n) creates a list from the values in the nth row of a text file (with values separated by semi-colons, tabs, or spaces, but not commas). E.g., **LOOKUP(3;FILEROW(myfile.txt;4))** will look up the third value in the 4th row of data.

FILECELL(filename;r;c) reads the value in column c and row r in a text file. E.g., **FILECELL(myfile.txt;4;3)** gives the same value as **LOOKUP(3;FILEROW(myfile.txt;4))** and **LOOKUP(4;FILECOL(myfile.txt;3))**

ILIST1, **ILIST2**, etc. create a list of the values of the ISvar for all living individuals in the population. (When used in a GSvar transition, the list is across all individuals in the metapopulation.) This allows the user to do such things as determine at what percentile an individual's ISvar falls in the population. Be careful to distinguish the meaning of **ILIST** from **ISHIST** and **ISLIST** (below). Similarly, be careful with **PLIST**, **PSHIST**, and **PSLIST**.



As a minor quirk of the *VORTEX* program, you cannot use both **ILIST** and either **ISLIST** or **ISHIST** in the same scenario unless the **ILIST** is used only within functions that describe population values (e.g., PSvars, K, number supplemented, etc.) and not in functions that describe values that are used for individuals. However, using **ILIST** and either **ISLIST** or **ISHIST** within the same scenario is always risky, because often variables that appear to relate to population values (such as mortality rates, breeding rates, and dispersal rates) are applied to each individual. It is safe to use **ILIST** in defining a PSvar, and **ISLIST** or **ISHIST** elsewhere in the same scenario.

Note that if a list has no elements, it is replaced by a list with just the value of -1. Thus, if a function contained “**MEAN(ILIST1)**”, but the population was extinct, then **ILIST1** would have no elements and the mean cannot be calculated. To avoid this, the function would return a “mean” of -1, as well as a maximum of -1, minimum of -1, etc. You will need to think carefully about how functions will be evaluated if there are no individuals in the list.

Accessing histories of State Variables

The operations **GSHIST**, **PSHIST**, and **ISHIST** can be used to access the prior values assigned to State Variables. For example, **GSHIST1(3)** gets the third value assigned to GS1 by its Transition function during the iteration. **GSHIST1(0)** gets the value assigned to GS1 by its Initialization function. Similarly, comparable **PSHIST** and **ISHIST** expressions get the values assigned to PSvars and ISvars. These can be useful for determining, for example, what was the value of a PSvar three years earlier, with a function such as **GSHIST4(Y-3)**.

The functions **GSLIST**, **PSLIST**, and **ISLIST** will create lists from the history of the State Variables. For example, **GSLIST1** creates a list of all values that have been assigned to GS1, starting with the Initialization value. These lists can be used to determine, for example, if a state variable has ever been **0**, with **FIND(0;GSLIST1)>0**. Or the lists can be used to determine the maximum value that a State Variable has had, for example, with **HIGH(ISLIST3)**.

Do not use any of these operations in an Initialization fn for a state variable, because the State Variable histories will not yet exist. If an element in a history is requested that does not exist, e.g., **GSHIST1(3)** when GS1 has had assigned only two values, or **GSHIST4(1)** when there are only 3 GSvars, then the operation will return **0**.

Using Random Numbers in Functions

Random number generators can be used to create a wide variety of stochastic events (for example, a 5-year drought that occurs on average once every 30 years), but the proper use of these functions requires careful consideration of how the “seed” values (implicit, as in **RAND** and **NRAND**, or explicit, as in **SRAND** and **SNRAND**) determine when new random numbers are selected. Repeated calls to the random number return the same value if the same seed is specified. Random numbers produced with different, even sequential, seeds will not be correlated. The “unseeded” forms (**RAND** and **NRAND**) set their own unique (or nearly so) seed each time they are called. The very first use of a random number generator in *VORTEX* uses a seed based on the number of seconds elapsed since the turn of the century. Each call to an unseeded random number generator also sets a new seed for the next call for an unseeded random number. Thus, identically configured computers starting the same simulation at exactly the same second on their clocks would produce identical results for an analysis. This synchrony may require, however, that all memory storage locations (including hard disk caches) and even the hard disk contents are identical on the systems (because they will affect the time required for each read or write to the disk).

The specification of random number seeds allows synchronization of sequences of random numbers. This can be used to create synchrony of events, such as catastrophes or environmental variation across populations, or autocorrelations among years (time lags or cycles). If several different demographic rates are specified by functions containing random number generators (perhaps to trigger separate catastrophes impacting survival and fecundity), care must be taken to create the desired synchrony or lack of synchrony. If two functions contain the same seed values, they will return the same random number. Seed values must be distinct to create independence of

random numbers. Proper use of random number seeds can be difficult. Think carefully about the effect of any seed that you use in a function, to be certain that it will produce the same random numbers when you want them, and independent random numbers otherwise. Any variable (e.g., **A** for age, **Y** for year, **R** for run, **P** for population) included within the seed will cause the same “random” number to be chosen for each case with the same value for those variables (**A**, **Y**, **R**, **P**). For example, if you specify **SRAND(P)** within a function, then each population will get an independent random number, and that set of random numbers will be the same over all calls to evaluate that function (such as for every year, every run, and every individual within each population). If you specify **SRAND((P*100)+Y)**, then each population will get a new independent random number each year of the simulation, but the set of random numbers will be the same across all runs of the simulation. You would normally want to include the variable **R** in the random number seeds (e.g., **SRAND((R*10000)+(P*100)+Y)**), in order to cause the random numbers to be independent among runs. See the examples below for further information about random number seeds.

The seeds used by *VORTEX* will be converted to integers between 0 and 65536. Non-integer seeds will be truncated [hence, **SRAND(35.23) = SRAND(35.89)**] and values above 64K will be “wrapped” [the modulus taken, so that **SRAND(65636) = SRAND(100)**].

Notes Regarding Function Syntax and Use

- Variables of trigonometric functions are assumed to be in radians, except for the function **RADIANS**, which returns a value in radians.
- The operator **NEG** is the same as using a minus (-) sign before a number. By the context, *VORTEX* will interpret whether a minus sign signifies subtraction (a binary operation) or the negative (a unary operation).
- **CEIL**, **FLOOR**, and **ROUND** convert rational numbers to integer values, but all expressions are evaluated as rational numbers. For example, **FLOOR(3.7)/FLOOR(4.1) = CEIL(2.1)/CEIL(3.7) = ROUND(3.1)/ROUND(3.6) = 0.75**.
- Numbers may be written with or without leading and trailing zeroes. Decimal points for integral values are optional. For example, all of the following are valid expressions (in the USA; see the next point): **3**; **3.00**; **0.03**; **.03**; **-0.30**; **-5**.
- Note, however, that the decimal delimiter (point or comma) used by Vortex will be the appropriate one for the regional data format set in Windows on the computer running *VORTEX*. Thus, $\frac{1}{4}$ would be written as **0,25** in most of Europe, but as **0.25** in the USA.
- Functions containing invalid mathematical expressions are prohibited, such as:

| | |
|-----------------------|--|
| SQR(-10) | Square root of a negative number |
| LN(-10) | Natural log of a negative number or zero |
| 5/0 | Division by zero |
| TAN(1.5707963) | Tangent of PI / 2 |
| ASIN(1.1) | Arcsine or arccosine of a value > 1, or < -1 |
- Some mathematically valid functions would be ambiguous or meaningless. For example, functions of carrying capacity (**K**) should not contain **K** as an independent predictor (of itself). Functions of **K** should also not include **A** (age) or **S** (sex) as parameters, because the condition of exceeding the carrying capacity is a population-level phenomenon, and **K** is

assessed once for each population each year. If **K** is a function of inbreeding (**I**), the value of **I** applied in the function will be the mean for the population.

- There is no limit on the length of a function. Often there are multiple functions that will achieve the same purpose. A more concise function may be easier to type and may also run faster in the program. However, it may be easier to see the logic in a more explicit but longer function that evaluates to the same result.
- Many of the variables that can be used in rate functions will themselves change during each year of the simulation. In order to avoid irresolvable circular interdependencies of parameters and rates, the population size (**N**), sizes of subsets (**J**, **F**, **M**, **U**, **X**, **W**), and genetic descriptors of populations (**G**, **I**, and **GDIST**) used in functions are updated at the beginning of each life history stage (e.g., Breed, Mortality, Disperse, Harvest, Supplement, etc.) for each population in the model. For example, the population size (**N**) at throughout Mortality calculations will be the value arising from the prior Breeding cycle, and the population size (**N**) used in Supplementation will be the value after the prior Harvest (assuming the default sequence of events in the year).

Even without specifying rates as functions, many of the rates used in *VORTEX* can be specified to be different for different years, sexes, ages, or inbreeding levels. (e.g., age-specific mortality, inbreeding depression in juvenile mortality, linear trends in K, etc.) Be aware that the effects of any functions entered are imposed on top of such dependencies that might be given in the standard input format. For example, EV in carrying capacity could be specified via standard input, or via a function of the type

$$K = 100 + (10 * NRAND),$$

thereby giving an annual level of EV in K equal to a standard deviation of 10.

An advantage of creating EV by specification within functions (rather than more simply as a parameter given to *VORTEX*) is that you have greater control over how EV is implemented in the model. For example, it is possible to specify that EV is concordant between two populations (but not with others):

$$\text{RATE} = 50 + [10 * SNRAND(Y + (R * 100) + [(P > 2) * 100 * SRAND(P)])].$$

In this function, the overall mean demographic rate is 50 with annual fluctuations due to EV (SD) equal to 10. Populations 3 and greater will experience independent annual fluctuations, while populations 1 and 2 will fluctuate synchronously. The use of year (**Y**) in the seed for the random number causes a new random number to be used each year. The use of **R** (iteration or run) in the seed causes the sequence of seed values to be different in each simulation. The inclusion of **(P>2)*100*SRAND(P)** within the seed causes a different sequence of random numbers to be chosen for each population after the first two have been evaluated. The seeds must include **Y+(R*100)** to ensure that every year-iteration is independent. (If you use **Y+R** as the seed, then year 3 of iteration 1 will have the same value as year 2 of iteration 2, etc.) In the simpler example of EV in K given above, no seed was needed or specified, so an independent random number will be selected each iteration, each year, and each population. These examples show how elaborate and non-intuitive the functions can become when you want to create even moderately complex models of population dynamics.

Using Functions to Examine Genetic Evolution

The parameters available for use in functions defining demographic rates include an individual's paternally inherited allele (**V**) and the maternally inherited allele (**Z**) of the (normally) non-selected locus, which is monitored for tracking genetic diversity. (The symbols for these variables **V** and **Z** have no intuitive meaning, but are rather the result of few letters remaining available for denoting additional parameters for functions.) By specifying that demographic rates are functions of the alleles carried by an individual, it is possible to model a wide variety of genetic processes impacting population dynamics, including: the effect and fate of alleles that confer alternative life history strategies (e.g., lower fecundity but higher survival); balancing, disruptive, or directional selection for alleles impacting demography; hybrid vigor or outbreeding depression caused by introgression of alleles from a distinct taxon or geographic population; and genetically based individual variation in demographic rates.

When the initial population is created, and when the population is supplemented with any new individuals, the founders are assigned unique alleles sequentially. Hence the first individual of the Population 1 is assigned alleles 1 and 2, the second individual is assigned alleles 3 and 4, and so on. Final frequencies of all founder alleles in each population, averaged across all iterations, can be outputted to a file if that option is selected on the Genetics input page. The allele frequencies are placed into a file with extension **.gen**.

Using VMacros in Vortex

VORTEX can call a macro file (a “script”) to process state variables in each year of the simulation. You specify a VMacro as a step in the annual sequence of events. This VMacro can update GSvars, PSvars (for every population), and IS vars (for every individual). This allows more complex, multi-step algorithms to be applied than would be available via one or more transition functions in the State Variable specifications. Macros can also use IF, ELSE, DO, WHILE, and other flow statements to create small programs for manipulating state variables. (See Evaluator.doc for information about macro syntax, and see the macro examples in the Vortex10Projects\Samples folder.)

Also, an input variable in *VORTEX* can be assigned a macro, just as it can be assigned a function, e.g., Rate =**MACRO**(“mymacrofile.txt”), or, equivalently, =**VMACRO**(“mymacrofile.txt”). The macro filename should be in quotes, and should include the path unless the path is the project folder. When the macro is evaluated, the return value that is assigned to the rate is the one identified by the RETURN statement of the macro. If no variable (or function) is given in the RETURN, then the last variable assigned a value in the macro is returned (but it would be a bad practice to rely on this rather than making explicit what variable gets returned). Although a rate can be set with a VMACRO, just as it can be with a function, a VMACRO cannot be embedded within a function.

It is important to be aware that there are a few subtle differences in how macros work in *VORTEX* and in *METAMODEL MANAGER*. This is because *VORTEX* automatically has an array of variables available for access in functions and macros, and automatically knows what state variables exist within the project; whereas *METAMODEL MANAGER* only knows what shared variables have been defined by its linked programs.

When VMacros are evaluated, all of the variables available in *VORTEX* functions will contain their current values. E.g., A, F, M, N, BROODS, etc. will all be read into the macro as input. If these variables are assigned a different value within the macro, they will retain that value only until the macro exits. I.e., except for state variables (see below) the changed value is not then passed onto the next function or macro that is evaluated.

When used within *VORTEX*, GSvars are “shared” in all macros, meaning that any change to their values in the macro get transferred back into the variable value. PSvars are “shared” in Population and Individual macros, but not in Global macros (because it would not be known to which population the value gets assigned). ISvars are not shared in either Global nor Population macros, because those macros cannot know to which individual the ISvar was referring. ISvars that are changed within a macro might not be passed back to the model (unless they are the RETURN variable), so they should be explicitly listed as SharedIndVars if the changed values need to be retained outside of the macro. This all means that the user needs to be careful about which variables within a macro are assigned values within the macro and which read initial values from the current values in the scenario. For example, if PS1 is used in a global macro, then it will not be assigned a predictable value from the scenario (as the population referenced is

undefined), but can be (and almost always should be) assigned value within any global macro in which it is used. (It would be even better to never use a PSvar within a global macro, or an ISvar within a population or global macro.)

The “Shared” variables and their initializations (“Vals=” statements) used in MMMacros for *METAMODEL MANAGER* are therefore less important in *VORTEX* Macros. Those lines can be left in the macro file – both so that the macro is compatible with *METAMODEL MANAGER* and so that those initial values can be used within the *MACROENTRY* program to test macros, but the lack of a Shared specification does not preclude the passing of the state variables as described above. The state variables in a *VORTEX* project are always “shared” in *VORTEX* macros – that is, the values of the state variables are available at the start of the macro, even if not explicitly “shared” in your macro script.

Note that if you use a macro to assign values to state variables, you will likely want to set the Transition function for the state variable to be the variable itself (e.g., “=GS1”). Otherwise, the transition will over-write changes that the macro made to the state variable.

Variables in the lists PPS, GSHIST, PSHIST, and ISHIST can be accessed within a VMacro, but their values cannot be changed, even within that VMacro. If you try to assign a new value to one of these variables, it will be treated as a new internal variable created within the macro, and it will not access the current value from the scenario.

VMacro has three commands that are not available in other programs (such as *METAMODELMANAGER*) that use mostly the same macro syntax. **EXPORT(“filename”)** creates a text file for each population that contains a complete listing of GSvars, PSvars, and all living individuals (including sex, age, damID, sireID, ISvars, and alleles). **IMPORT(“filename”)** reads data back from a text file for each population that contains a complete listing of GSvars, PSvars, and all living individuals (including sex, age, damID, sireID, ISvars, and alleles). The filename should be in quotes, so that any punctuation in the filename doesn’t get misinterpreted as a delimiter between function parameters. If the filename contains the path (e.g., “C:/MyDocs/myfile”), that path will be used, but the slashes between levels should be forward slashes (‘/’), not backward slashes (‘\’). If the filename does not contain the path, the file will be placed in the VOutput subfolder of the project folder. The filename should not normally include an extension, because a “.txt” extension will be added by *VORTEX* if it is not already there. The **IMPORT** command uses the same rules regarding the filename.

By issuing an **EXPORT** command, then running an external program (such as an R script, see below) to modify the data, and then issuing an **IMPORT** command, external programs can modify the *VORTEX* population during the simulation. The **PAUSE** command exports the data file, stops and asks the user when to proceed, and then imports the data file back in. This allows the user to externally examine or manipulate the data file during the simulation. (This works the same as a Pause step in the annual sequence of events or a step-wise run of the scenario, except that it allows the user to Pause the simulation in the middle of a macro.)

Macros and functions are powerful, so use them carefully and test them thoroughly!

VORTEX installs a program *MMMMACROENTRY* (with a shortcut placed in the Vortex10Projects folder) that allows you to test macros to see if they are functioning the way you intend.

Macros have been rather lightly tested in *VORTEX*, so let us know (help@scti.tools) if you encounter problems when using them.

Using R scripts in Vortex

Functions and Macros in *VORTEX* can now call R scripts to process data and provide results that can be used in input or as output descriptors. When any R script or command is run within a function or macro, an instance of the R engine will be started if one was not already started by the program. The same R engine remains in use (i.e., it is not closed and then reopened) until the *VORTEX* program closes. This means that a script can be run to do some calculations, and then the working environment and all of its data remain accessible to further R scripts.

Methods available in functions and Macros for running R scripts:

RMACRO(rscriptfile) -- runs the R script in the file. Note that normally this is where you would put a command to load an R library to be used for later commands. In here, you can also put other start-up commands that might be needed. Rscriptfile must be passed as a string in quotes, e.g., **RMACRO("C:/WorkingFolder/myRfile.txt")**. If the path is not provided, *VORTEX* will assume that the file is in the project folder. The user would normally need to run such a script when a scenario starts running, in order to load the necessary R libraries. Therefore, the GUI now provides on the bottom of the Scenario Settings tab a place to enter such an R script file that will then be run (via an **RMACRO** statement) when the scenario is started.

REVAL(command) – evaluates the R command. The command must be a quoted string, e.g., **REVAL("newdata<-data.frame(A=3,B=4,C=5)")**, in order to avoid confusion between characters in the R command and delimiters and operators in the function. Returns a 1, or a 0 on error.

REVALN(command) – equivalent to **REVAL**, except that it returns a numeric value returned by the R command. Crashes if R does not return a value that can be represented by a number.

REVALGS(command) – creates a data.frame called RDATA with all the current GSvars and the variables A to Z, and then evaluates the R command. The data.frame labels the variables by their labels, not by GS1, GS2, etc. (unless those are also used as the labels). Labels will be given in upper case when assigned to RDATA, so reference to them in the R command must also use upper case. Returns a 1, or a 0 on error.

REVALGSN(command) – equivalent to **REVALGS**, except that it returns a numeric value returned by the R command. Crashes if R does not return a value that can be represented by a number.

REVALPS(command) – creates a data.frame (RDATA) with the current GSvars, PSvars, and variables A to Z, and then evaluates the R command. Returns a 1, or a 0 on error.

REVALPSN(command) – equivalent to **REVALPS**, except that it returns a numeric value returned by the R command. Crashes if R does not return a value that can be represented by a number.

REVALIS(command) – creates a data.frame (RDATA) with the current GSvars, PSvars, ISvars, and variables A to Z, and then evaluates the R command. Returns a 1, or a 0 on error.

REVALISN(command) – equivalent to **REVALIS**, except that it returns a numeric value returned by the R command. Crashes if R does not return a value that can be represented by a number.

REVALVARS(command) – creates a data.frame (RDATA) with the current variables A to Z (plus any further VAR1, VAR2, etc., but most people don't know that these extra variables beyond A to Z exist in Vortex) and then evaluates the R command. Returns a 1, or a 0 on error.

REVALVARSN(command) – equivalent to **REVALVARS**, except that it returns a numeric value returned by the R command. Crashes if R does not return a value that can be represented by a number.

REVALVALS(command; x1; x2; x3; ...) – creates a data.frame (RDATA) with the values x1, x2, x3, etc. and then evaluates the R command. The data.frame labels for these variables will be VAR1, VAR2, VAR3, etc. Returns a 1, or a 0 on error.

REVALVALSN(command; x1; x2; x3; ...) – equivalent to **REVALVALS**, except that it returns a numeric value returned by the R command. Crashes if R does not return a value that can be represented by a number.

When writing the R commands above, the user will need to be cognizant that the data.frame created by the above commands is always called RDATA, although other data.frames can be created with R commands executed with **REVAL** or **RMACRO**.

A few more things to be aware of ...

- R commands are case sensitive, whereas functions are not. E.g., **reval("GSDATA<-RDATA")** is the same as **REVAL("GSDATA<-RDATA")**, but these are not the same as **REVAL("gsdata<-rdata")**.
- R commands are passed as quoted strings to **REVAL**, e.g., as **REVAL("GSDATA<-RDATA")**, *not* as **REVAL(GSDATA<-RDATA)**, and this means that the R command can contain symbols that would normally cause problems within a function (e.g., might be interpreted as operators or delimiters) if they were outside of a quoted string. (E.g., in a function, GSDATA<-RDATA, without quotes, would be a test of whether a variable called GSDATA was less than the negative of a variable called RDATA.) When an R command contains elements that are in quotes, you must use single quotes rather than double quotes within the R command, so that the (single!) quotes within the R command do not get confused with the (double!) quotes that encase the R command.

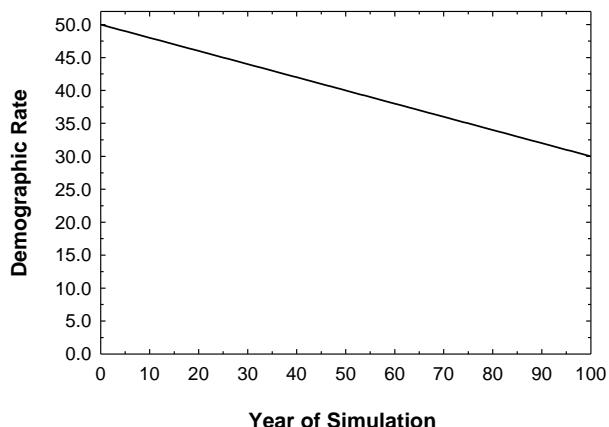
Examples of Rate Functions

The easiest way to demonstrate the formats in which functions can be entered into *VORTEX* is with a series of examples. The examples shown below include a plot of the function where appropriate as well as the actual expression.

1. Continuous linear decline over time

$$\text{RATE} = 50 - (0.2 * Y)$$

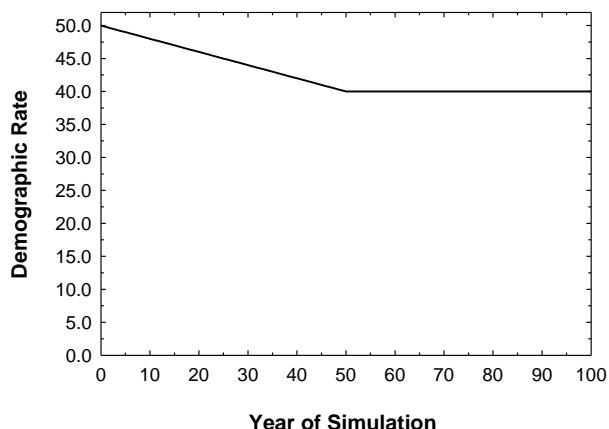
This function specifies a starting rate (perhaps for adult female breeding success or for carrying capacity) equal to 50 in year 0, with a decline of 0.2 per year resulting in a rate equal to 30 after 100 years.



2. Linear decline limited to a period of years

$$\text{RATE} = 50 - (0.2 * \text{MIN}((Y-1); 50))$$

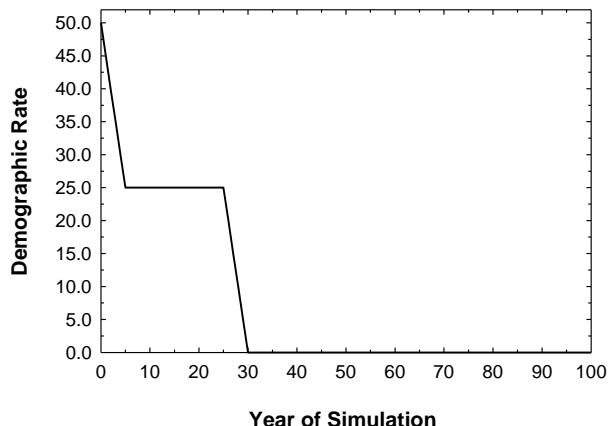
In this case, the decline occurs only through the first 50 years of the simulation. Note that the decline is specified to start in year 2, so that year 1 still has a rate of 50. This is the form of the function used by *VORTEX* if the user specifies a linear trend in carrying capacity.



3. Linear decrease during intervals of years

$$\text{RATE} = 50 - (5 * (\text{MIN}(5; Y) + ((Y-25) * (Y > 25))))$$

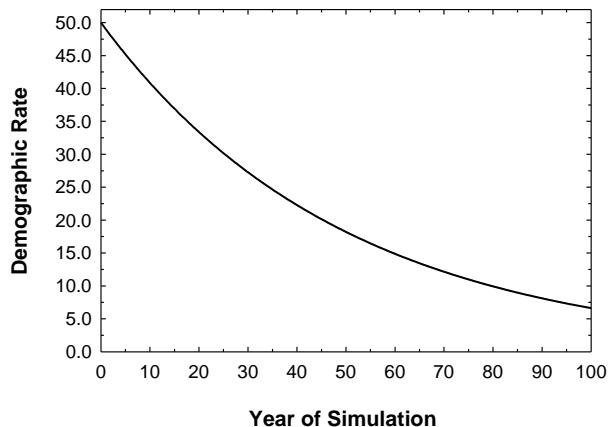
The rate starts at 45 in year 1, declines to 25 by year 5, and again resumes the decline at a rate of 5 per year after year 25.



4. Exponential decline

$$\text{RATE} = 50 * (0.98^Y)$$

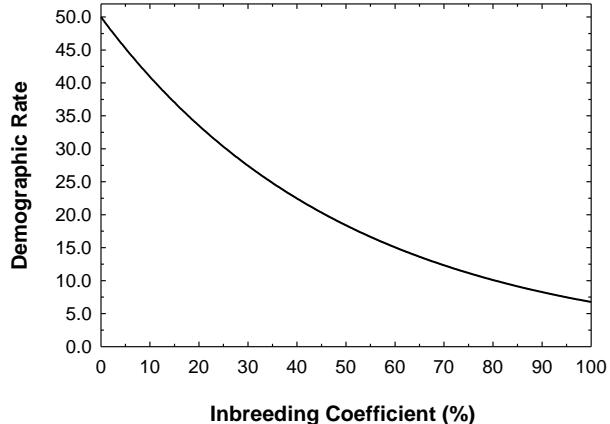
The rate declines by 2% each year.



5. Exponential decline with inbreeding

$$\text{RATE} = 50 * \text{EXP}(-2.0 * I)$$

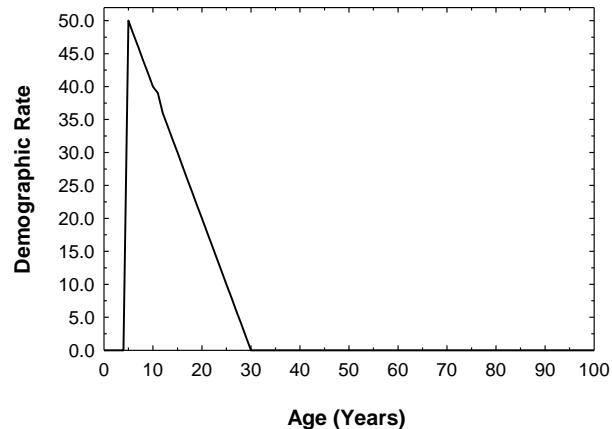
The rate declines from 50 in non-inbred animals down to 6.7 in fully inbred animals ($I = 1$). An equation like this might be used to specify a decline in fecundity due to inbreeding.



6. Age-dependent fecundity, with linear decline after the onset of breeding

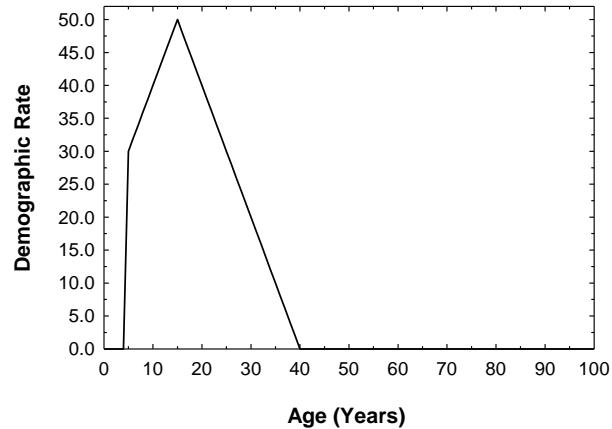
$$\text{RATE} = (\text{A}>=5)*(50 - ((\text{A}-5)*2))$$

Breeding begins with a rate of 50 at age 5, but then declines by 2 each year thereafter.



7. Age-dependent fecundity, with a symmetrical peak at age 15

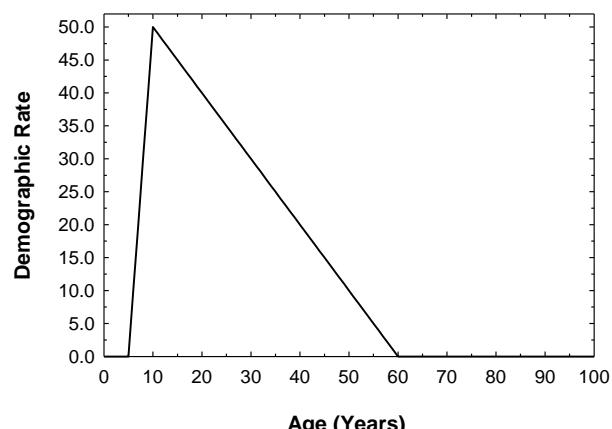
$$\text{RATE} = (\text{A}>=5)*(50 - (\text{ABS}(\text{A}-15)*2))$$



8. Age-dependent fecundity, with an asymmetrical peak at age 10

$$\text{RATE} = (\text{A}>=5)*[\{\text{[A}<\text{10}]*([\text{A}-\text{5}]*10)\} + (\text{[A}>\text{9}]*[\text{60}-\text{A}])]$$

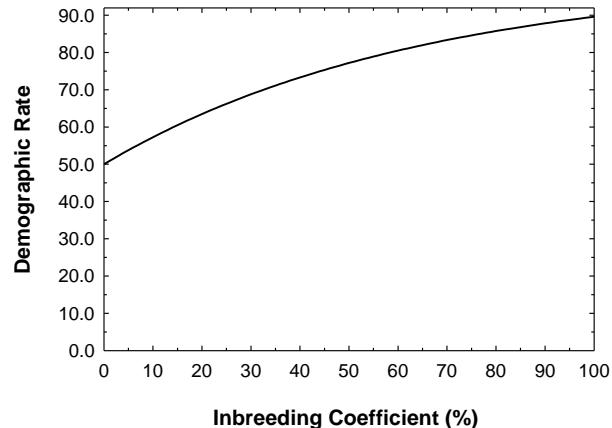
Different trends are specified for age intervals 0-4, 5-9, and 10+. Note the use of parentheses, brackets, and braces to improve readability.



9. Increase in mortality with inbreeding

$$\text{RATE} = 100 - (50 * \text{EXP}(-1.57 * I))$$

The survival rate declines exponentially (described by the portion within the outermost parentheses), while the percent mortality is set at 100-survival. This is the equation used by VORTEX to model increased juvenile mortality if there are 3.14 lethal equivalents and no recessive lethal alleles contributing to inbreeding depression.

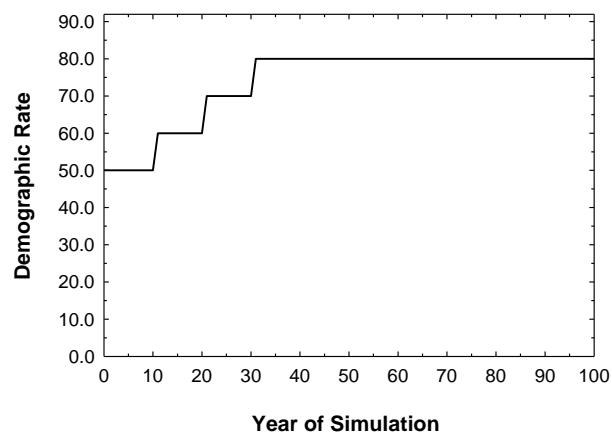


10. Stepwise increase

$$\text{RATE} = 10 + ((Y > 10) * 10) + ((Y > 20) * 10) + ((Y > 30) * 10)$$

The rate increases from 50 to 80 at 10-year intervals. An alternative way to express the same function would be

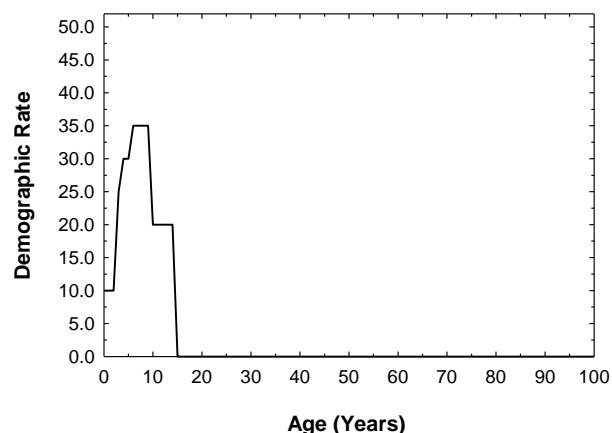
$$\text{RATE} = 50 + (10 * (\text{MIN}(3; \text{FLOOR}((Y-1)/10))))$$



11. Different rates at different intervals

$$\text{RATE} = \\ (10 * (A < 3)) + (25 * (A = 3)) + (30 * ((A = 4) \text{ OR } (A = 5))) + (35 * ((A > 5) \text{ AND } (A < 10))) + (20 * ((A >= 10) \text{ AND } (A < 15)))$$

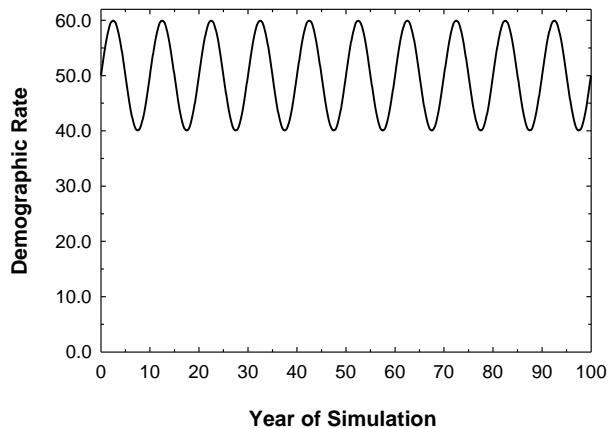
The rate increases stepwise with age, then drops to a lower level for years 10 through 14, and then drops to zero for animals 15 years and older. Note that although it can be tedious, any rate function can be modeled by specifying the rate for each interval of the dependent variable.



12. Cyclical response

RATE = 50+(10*SIN(PI*Y)/5)

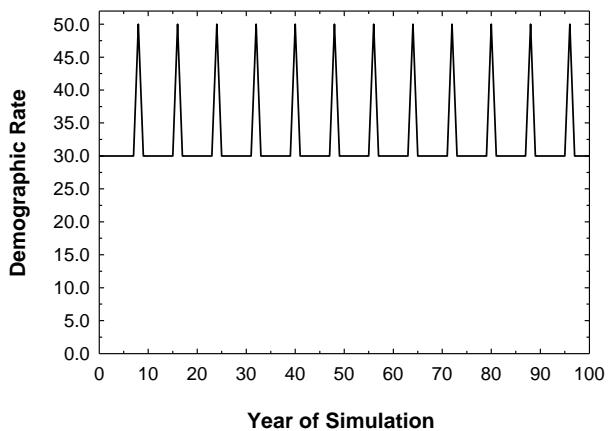
Here the rate fluctuates between 40 and 60 according to a sine wave with a 10-year periodicity.



13. Regular pulses of a higher rate

RATE = Y%8=0*20+30

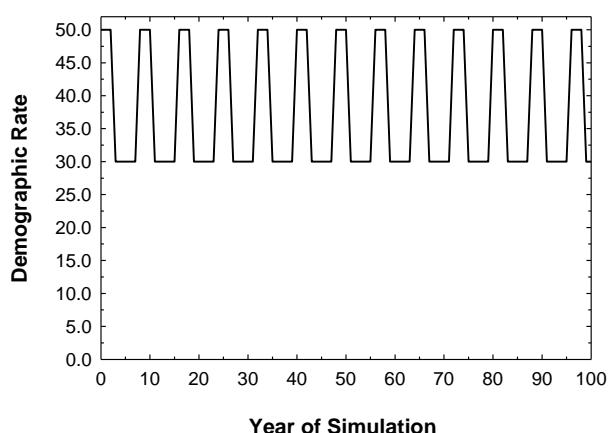
Background rate of 30 jumps to 50 every 8th year. Note that the order of operators was left to the *VORTEX* default (left to right). This is not a safe practice, but it does work in this case.



14. Pulses of longer duration

RATE = (((Y%8)<3)*20)+30

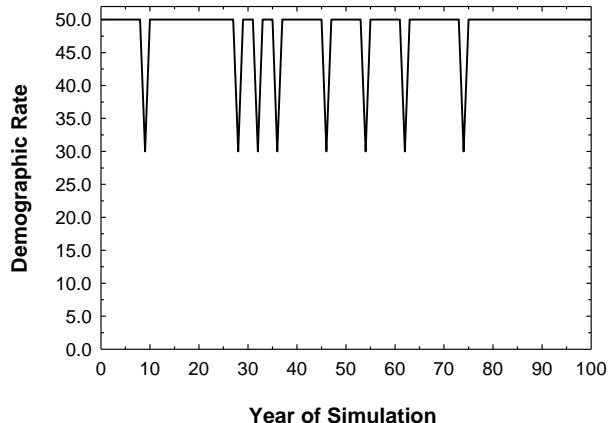
The rate jumps to 50 for a 3-year time span every 8th year. In this case, parentheses were used (wisely) to be sure that the intended order of operators was followed.



15. Random pulses: catastrophes

RATE = $50 - (20 * (\text{SRAND}(Y + (R * 100)) < 0.05))$

The background rate of 50 drops to 30 on average once every 20 years. A seeded random number is needed; otherwise the years in which the rate drops would be independent among individuals (effectively, the rate would continuously be 49). The seed of $Y + (R * 100)$ causes a different seed to be used for each year of each iteration (if there are 100 or fewer years). The above function is equivalent to specifying a catastrophe with frequency = 5% and severity = 0.60.



16. Random pulses independent among populations

RATE = $50 - (20 * [(\text{SRAND}[Y + (R * 100)] + (100 * \text{SRAND}(P))] < 0.05))$

The “catastrophes” are independent among populations, because each population (P) sets a new (and random) seed for the random number generator which tests whether the catastrophe occurs. Careful use of parentheses (or brackets) is critical in this function, in order to ensure that the random number seeds work as intended.

17. Catastrophes affecting only selected age class(es)

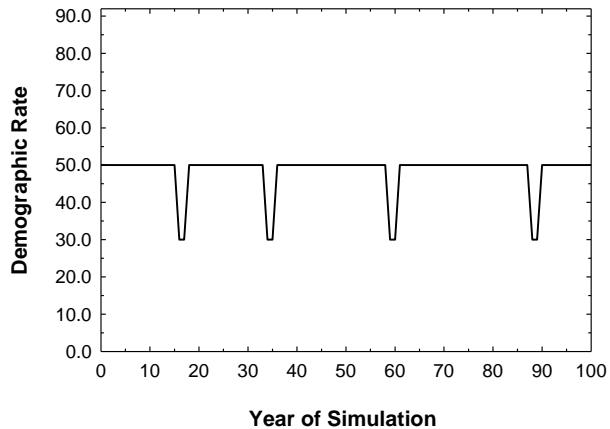
RATE = $50 - ((A < 3) * 20 * (\text{SRAND}(Y + (R * 100)) < 0.05))$

The catastrophe affects only individuals of ages 1 and 2. The 2-D graphs of this function do not illustrate the age-dependent relationship: The graphs against Y and R set $A = 1$ (and show the catastrophes affecting young individuals); while the graph against A happens to display a year and iteration ($Y = R = 1$) in which no catastrophe for any age occurs.

18. Multi-year catastrophes

RATE = $50 - (20 * (\text{SRAND}((Y/2) + (R * 100)) < 0.10))$

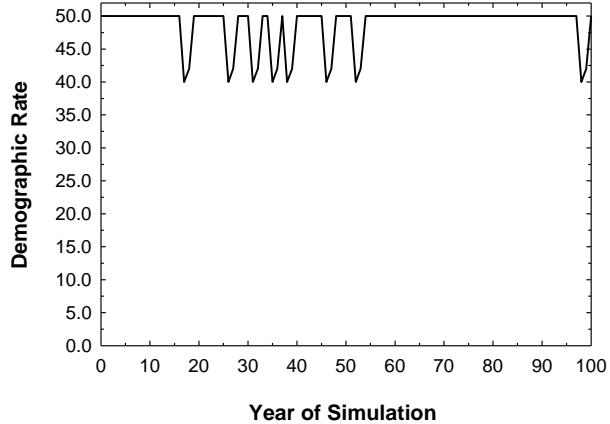
The catastrophes have a 2-year impact, because the seed value is converted to an integer, giving pairs of subsequent years the same random number. The frequency per year is 10%, so that the frequency of an onset of a 2-year catastrophe is 5%.



19. Multi-year catastrophes with a decreased impact in year 2

$$\text{RATE} = 50 - ([10 * (\text{SRAND}(Y + (R * 100)) < 0.05)] + [8 * (\text{SRAND}((Y - 1) + (R * 100)) < 0.05)])$$

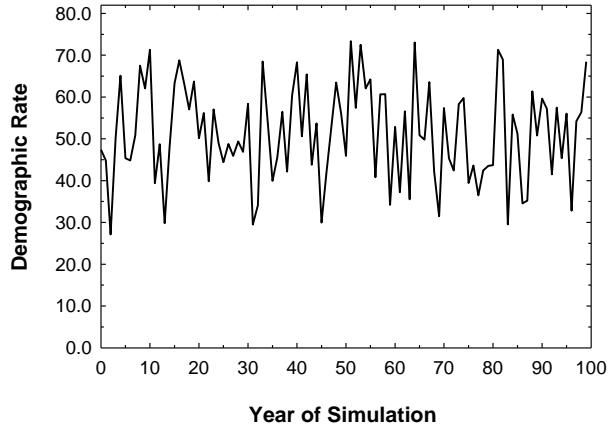
The second seed is the same as the first seed from the previous year. Thus the catastrophe has a lesser impact (severity = 0.84 rather than 0.80) in the second year. This approach can also be used to model catastrophes which impact survival in one year (using a function with an expression like that in the first brackets above) and fecundity in the second year (using a function containing the expression in the second set of brackets). Note that in example #18, catastrophes always start in even-numbered years, while in this example catastrophes can begin in any year.



20. Random variation across years

$$\text{RATE} = 50 + [10 * (\text{SNRAND}(Y + (R * 100)))]$$

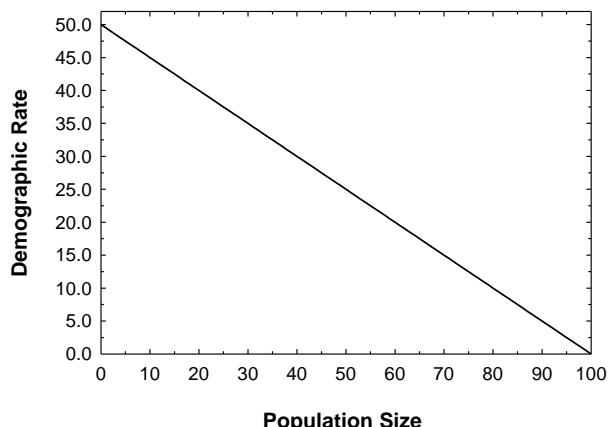
This is the same as imposing a mean rate of 50 with environmental variation of SD = 10.



21. Linear density dependence

$$\text{RATE} = 50 * ((K - N) / K)$$

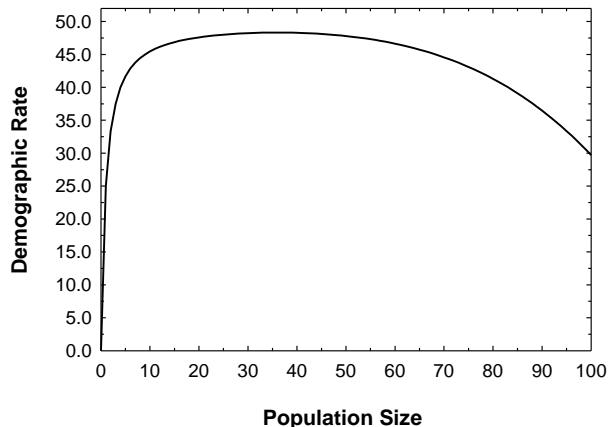
The rate declines from 50, at $N = 0$, to 0 when $N = K$.



22. Density dependence used as the default for breeding in VORTEX

$$\text{RATE} = [50 - (20 * [(N/K)^4])] * [N / (1 + N)]$$

The rate peaks near 50 when N is small, declines at higher densities, and is 30 when $N = K$. At very small N , the rate is also depressed. For example, it is reduced by 50% when $N = 1$, and reduced by 25% when $N = 3$. In terms of the coefficients that can be entered into the optional density-dependence for breeding in VORTEX, $P(0) = 50$, $P(K) = 30$, $B = 4$, and $A = 1$.



23. Sex-specific dispersal rates

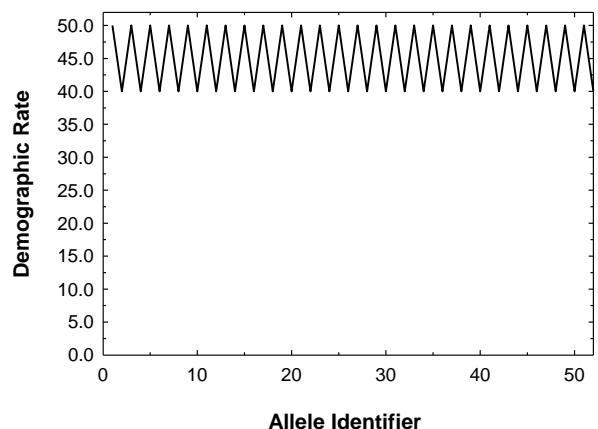
$$\text{RATE} = D * [(S='M') \text{OR} (\text{RAND}>0.35)]$$

If the above function is used for the Dispersal Modifier Function, then 35% of the females are prevented from dispersing. Thus, dispersal rates for females are effectively reduced by 35% relative to male dispersal. An unseeded random number is used so that dispersal will be determined independently each female. Note that the dispersal rates entered subsequently (D) will be those applied to males, with females having lower rates. A similar approach can be used to create age-specific dispersal rates (or dispersal mortality).

24. Alleles confer differential reproductive rates

$$\text{RATE} = 40 + (10 * (V\%2)) + (10 * (Z\%2))$$

In this case, half of the alleles (specifically, those with even numbers) cause an increment of 10 in the breeding rate of their carriers. An individual that is homozygous for an even-numbered allele will have a breeding rate equal to 60%, while those homozygous for an odd-numbered allele will have a rate equal to 40%.



25. Overdominance for survival, all unique founder alleles

$$\text{RATE} = 20 + (10 * (\text{V}=\text{Z}))$$

An infinite alleles model in which homozygotes have a mortality of 30%, while the rate for heterozygotes is 20%.

26. Overdominance for survival, two functionally distinct founder alleles

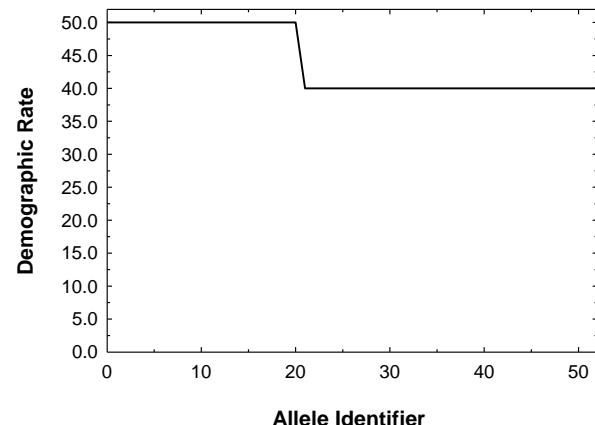
$$\text{RATE} = 20 + (10 * ((\text{V}\%2)=(\text{Z}\%2)))$$

A two allele model in which homozygotes with two odd or two even alleles have a mortality of 30%, while the rate for heterozygotes is 20%.

27. Outbreeding depression for breeding rate upon introgression from supplemented individuals

$$\text{RATE} = 50 - (10 * ((\text{V}<20)=(\text{Z}>19)))$$

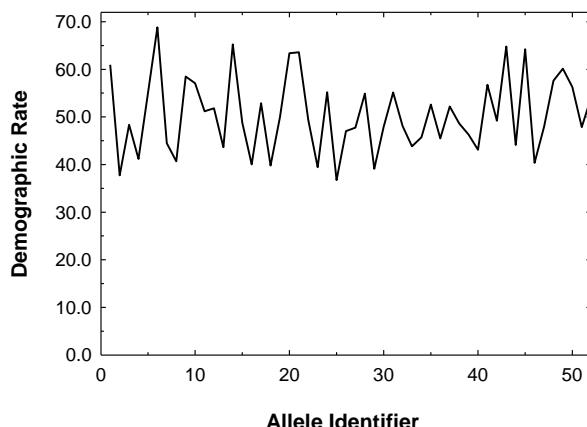
With ten initial founders, and with some number of individuals from another source used as supplements at a later stage, the breeding rate is 50% for an individual which carries both of its alleles from the initial founders, or both from the source population of the supplements, vs. 40 for individuals which are heterozygous, carrying an allele from each source.



28. Genetically-based individual variation in breeding success

$$\text{RATE} = 50 + (5 * (\text{SNRND}((\text{R}*100)+\text{V})+\text{SNRND}((\text{R}*100)+\text{Z})))$$

In this case, breeding rates vary around a mean of 50 with a standard deviation equal to $5 * \text{SQRT}(2)$.



Appendix 1.

An Overview of Population Viability Analysis Using *VORTEX*

This Appendix presents an overview of processes threatening the health and persistence of wildlife populations, the methods of population viability analysis, the *VORTEX* simulation program for PVA, and the application of such techniques to wildlife conservation. Much of the following material is adapted from Lacy (1993a) and Lacy (1993/4). This description was first written to describe *VORTEX* version 9. Please see the [overview of changes](#) from *VORTEX* 9 to *VORTEX* 10 (below) for a summary of new options in *VORTEX* 10.

The Dynamics of Small Populations

Many wildlife populations that were once widespread, numerous, and occupying contiguous habitat have been reduced to one or more small, isolated populations. The primary causes of the decline of many species are obvious and deterministic: Populations are over-harvested; natural habitat is converted and lost to the species, often involving the replacement of diverse ecological communities with monocultures; environments are polluted, with the dumping of toxins into the air, water, and soil; local and now even global climates are modified by the actions of humans; and numerous exotic competitors, predators, parasites and diseases are introduced into communities that have never evolved defenses to the new invaders. The primary causes of species decline are usually easy to understand, and often easy to study and model, but usually, though not always, difficult to reverse. Even if the original causes of decline are removed, a small isolated population is vulnerable to additional forces, intrinsic to the dynamics of small populations, which may drive the population to extinction (Shaffer 1981; Soulé 1987; Clark and Seebeck 1990).

Of particular impact on small populations are stochastic, or random or probabilistic, processes. Indeed, the final extinction of most populations often occurs not so much because of a continuation of the pressures that led to the initial decline, but because of bad luck. Chance, or stochastic, processes usually have little impact on long-term population dynamics, as long as the population is abundant and spread over a wide geographic range and a number of habitats. Deterministic processes, such as those listed above, predominate in widespread, still common species, while local chance events impacting subsets of the population will average out across the broader, diverse range. When a population becomes small, isolated, and localized, however, chance events can become important, even dominating the long-term dynamics and fate of a population.

Many stages in the life history of an organism, and the processes that define the history of a biological population, are essentially stochastic sampling phenomena. Births, deaths, dispersal, disease, sex determination, and transmission of genes between generations all are largely probabilistic phenomena. Small samples intrinsically have greater variance around the probabilistic mean or expectation than do large samples, and therefore small populations will experience greater fluctuations in births, deaths, sex ratio, and genetic variation than will larger populations. The fundamental problem facing small populations is that the fluctuations they experience due to the multiple stages of sampling each generation make it increasingly likely that the populations will, unpredictably, decline to zero. Once populations are small, the probability that they will become extinct can become more strongly determined by the amount of fluctuations in population size than in the mean, deterministic population growth rate. Thus, extinction

can be viewed as a process in which once common and widespread populations become reduced to small, isolated fragments due to extrinsic factors, the small remnant populations then become subjected to large fluctuations due to intrinsic processes, the local populations occasionally and unpredictably go extinct, and the cumulative result of local extinctions is the eventual extinction of the taxon over much or all of its original range (Gilpin and Soulé 1986; Clark et al. 1990).

The stochastic processes impacting on populations have been usefully categorized into demographic stochasticity, environmental variation, catastrophic events, and genetic drift (Shaffer 1981). Demographic stochasticity is the random fluctuation in the observed birth rate, death rate, and sex ratio of a population even if the probabilities of birth and death remain constant. Assuming that births and deaths and sex determination are stochastic sampling processes, the annual variations in numbers that are born, die, and are of each sex can be specified from statistical theory and would follow binomial distributions. Such demographic stochasticity will be most important to population viability perhaps only in populations that are smaller than a few tens of animals (Goodman 1987), in which cases the annual frequencies of birth and death events and the sex ratios can deviate far from the means.

Environmental variation is the fluctuation in the probabilities of birth and death that results from fluctuations in the environment. Weather, the prevalence of enzootic disease, the abundances of prey and predators, and the availability of nest sites or other required microhabitats can all vary, randomly or cyclically, over time. The fluctuations in demographic rates caused by environmental variation is additive to the random fluctuations due to demographic stochasticity. Thus, the difference between the observed variation in a demographic rate over time and the distribution describing demographic variation must be accounted for by environmental variation.

Catastrophic variation is the extreme of environmental variation, but for both methodological and conceptual reasons rare catastrophic events are analyzed separately from the more typical annual or seasonal fluctuations. Catastrophes such as epidemic disease, hurricanes, large-scale fires, and floods are outliers in the distributions of environmental variation. As a result, they have quantitatively and sometimes qualitatively different impacts on wildlife populations. (A forest fire is not just a very hot day.) Such events often precipitate the final decline to extinction (Simberloff 1986, 1988). For example, one of two populations of whooping crane was decimated by a hurricane in 1940 and soon after went extinct (Doughty 1989). The only remaining population of the black-footed ferret (*Mustela nigripes*) was being eliminated by an outbreak of distemper when the last 18 ferrets were captured (Clark 1989).

Genetic drift is the cumulative and non-adaptive fluctuation in allele frequencies resulting from the random sampling of genes in each generation. This can impede the recovery or accelerate the decline of wildlife populations for several reasons (Lacy 1993b). Inbreeding, not strictly a component of genetic drift but correlated with it in small populations, has been documented to cause loss of fitness in a wide variety of species, including virtually all sexually reproducing animals in which the effects of inbreeding have been carefully studied (Wright 1977; Falconer 1981; O'Brien and Evermann 1988; Ralls et al. 1988; Lacy et al. 1993; Lacy 1997). Even if the immediate loss of fitness of inbred individuals is not large, the loss of genetic variation that results from genetic drift may reduce the ability of a population to adapt to future changes in the environment (Fisher 1958; Robertson 1960; Selander 1983).

Thus, the effects of genetic drift and consequent loss of genetic variation in individuals and populations negatively impact on demographic rates and increase susceptibility to environmental perturbations and catastrophes. Reduced population growth and greater fluctuations in numbers in turn accelerates genetic drift (Crow and Kimura 1970). These synergistic destabilizing effects of stochastic process on small populations of wildlife have been described as “extinction vortices” (Gilpin and Soulé 1986).

What is Population (and Habitat) Viability Analysis?

Analyses which have used the *VORTEX* simulation for guiding conservation decisions refer variously to “Population Viability Analysis (PVA)”, “Population and Habitat Viability Analysis (PHVA),” “Population Vulnerability Analysis”, “Population Viability (or Vulnerability) Assessment”, and other variants on the name. This diversity of terminology has caused some confusion among practitioners of the PVA (or PHVA) approach, and probably even more confusion among wildlife managers who have tried to understand what analysis was being described, and whether it could be a useful tool in their efforts to conserve biodiversity. The diversity of perceptions about the PVA approach is not limited to its name. Different people mean different things by PVA, and the definitions and practice of PVA are constantly evolving. We don’t think it is not the case, as has sometimes been suggested, that some people are doing PVA correctly, and others incorrectly, but rather that people are using different (if related) kinds of analyses and labeling them with the same (or similar) terms. What analysis is correct depends on the need and the application. Below, we attempt to clarify what PVA is, by suggesting a more consistent terminology and by describing the features that characterize the application of the PVA approach to conservation. The perspective offered here is necessarily biased by personal experiences in conservation; we will not attempt an exhaustive historical account of this field.

Population viability analysis originally described methods of quantitative analysis to determine the probability of extinction of a population. Shaffer (1981) first defined a minimum viable population (MVP) as the size at which a population has a 99% probability of persistence for 1000 years, but it might be more meaningful biologically to consider it to be the size below which a population's fate becomes determined largely by the stochastic factors that characterize extinction vortices. One concept of population viability analysis is any methodology used to determine an MVP (Shaffer 1990). More broadly, PVA is the estimation of extinction probabilities and other measures of population performance by analyses that incorporate identifiable threats to population survival into models of the extinction process (Brussard 1985; Gilpin and Soulé 1986; Burgman et al. 1993; Lacy 1993/1994).

Shaffer's (1981) original term “minimum viable population” (MVP) has fallen into disfavor (Soulé 1987), even as the PVA approach has risen in popularity. Shaffer stressed that an MVP was an estimate of the population size below which the *probability* of extinction was unacceptably high, that different populations would have different MVPs, and that the MVP determined for a population would depend on the threatening factors that were considered. However, the term implied to some people that there was a well-defined number below which extinction was certain and above which persistence was assured. Re-emphasizing the probabilistic nature of the extinction process, a number of conservation biologists have focused on methods for estimating the probability of extinction over defined time periods for a designated population exposed to a specific scenario of environmental conditions, threats to persistence, and future management actions and other foreseeable events (Brussard 1985; Starfield and Bleloch 1986; Soulé 1987; Simberloff 1988; Gilpin 1989; Shaffer 1990; Boyce 1992; Burgman et al. 1993). Thus, “Population Viability Analysis” (or the synonymous “Population Viability Assessment” and “Population Vulnerability Analysis”) came to describe any of the array of methods for quantifying the probability of extinction of a population. Although PVA has been extended by some to encompass a broader approach to conservation (see below), the term “Population Viability Analysis”, or PVA, should perhaps be reserved for its original, yet still rather broad, meaning.

Beginning in about 1989 (Lacy et al. 1989; Seal and Lacy 1989; Seal et al. 1990), it became increasingly recognized that PVA can often be most usefully incorporated into a strategy for the conservation of a taxon if it is part of, and often central to, a conservation workshop that mobilizes collaboration among the array of people with strong interest in or responsibility for a conservation effort (e.g., governmental wildlife agencies, conservation NGOs, and the local people who interact with the species or its habitat) or with particular expert knowledge about the species, its habitats, or the threats it faces (e.g., academic

biologists, conservation professionals, other wildlife biologists, experts on human demographics and resource use). Conservation problems are almost always multi-faceted, involving not only complex dynamics of biological populations, but also interactions with human populations, the past, present, and future impacts of humans on habitats, and human political, social, and economic systems (Alvarez 1993; Bormann and Kellert 1991; Clark 1989, 1993). Many people need to contribute knowledge, expertise, and ideas in order to achieve the recovery of threatened species. Population viability analyses can provide a framework for incorporating the many needed kinds of knowledge into species conservation efforts, because PVAs do allow the assessment of many kinds of factors that threaten the persistence of populations (Lacy 1993a; Lindenmayer et al. 1993).

The Conservation Breeding Specialist Group (CBSG) of the IUCN's Species Survival Commission especially has advocated and used workshops centered on PVAs to provide guidance to conservation assessment and planning. Over the past few years, the PVA workshop as an approach to species conservation has expanded considerably beyond the quantitative analysis of extinction probabilities as advanced by Shaffer (1981, 1990), Soulé (1987), Gilpin (1989), Clark et al. (1991), Boyce (1992), and others. PVA workshops have incorporated consideration of resource use and needs by local human populations (Seal et al. 1991; Bonaccorso et al. 1999), education programs for the local human populations (Odum et al. 1993), trade issues (Foose et al. 1993), and trends in human demographics and land use patterns (Walker and Molur 1994; Herrero and Seal 2000). Recognizing that the conservation assessment workshops increasingly incorporated more than just the population biology modeling (which still formed a core organizing and analysis framework for the workshop), the CBSG has termed their workshops *Population and Habitat Viability Analyses* (PHVA). We would recommend that the term Population and Habitat Viability Analysis (PHVA) be used to describe the collaborative workshop approach to species conservation that centers on, but encompasses more than, a Population Viability Analysis (in the narrow sense). The concept of a PHVA continues to expand and evolve, as it should considering the need for more holistic and flexible approaches to conservation (e.g., Ruggiero et al. 1994). Thus, in the usage I recommend, PVA is a quantitative analysis of the probability of population persistence under defined sets of assumptions and circumstances. PHVA is a workshop process that brings to bear the knowledge of many people on species conservation, eliciting and assessing multiple options for conservation action, principally by using the tool of PVA as a way evaluate present threats to population persistence and likely fates under various possible scenarios.

Population Viability Analysis (PVA)

Two defining characteristics of a PVA are an explicit model of the extinction process and the quantification of threats to extinction. These features set PVA apart from many other analyses of the threats facing species, including, for example, the IUCN Red Books of Threatened Species. As a methodology to estimate the probability of extinction of a taxon, PVA necessarily must start with an understanding, or model, of the extinction process (Clark et al. 1990).

Generally, the model of extinction underlying a PVA considers two categories of factors: deterministic and stochastic. Deterministic factors, those that can shift species from long-term average population growth to population decline include the well-known threats of over-harvest, habitat destruction, pollution or other degradation of environmental quality, and the introduction of exotic predators, competitors, and diseases. Singly or combined, these forces have driven many wildlife populations to low numbers and, for some, to extinction. Once a population becomes small, and isolated from conspecific populations that might serve as sources for immigrants that could stabilize demographics and genetics, its dynamics and fate can become dominated by a number of random or stochastic processes (as outlined above and by Shaffer 1981). Thus, even if the original deterministic causes of decline are stopped or reversed, the instability caused by the action of stochastic processes acting on small populations can cause the extinction of a population.

In nature, most threatening processes have both deterministic and stochastic features. For example, a high level of poaching might be seen as a deterministic factor driving a wildlife population toward extinction, but whether an individual animal is killed might be largely a matter of chance. In a PVA, poaching might be modeled as a deterministic process by killing a determined proportion of the animals, or it might be modeled as a stochastic process by giving each animal that probability of being killed but allowing the exact numbers killed to vary over time. If the population is large and the percent of animals killed is high, then these two ways of modeling the effects of poaching will yield the same results: the deterministic component of poaching dominates the population dynamics. If the population is small or the percent of animals killed is very low, then the numbers killed in a stochastic model (and in nature) might vary substantially from year to year: the stochastic nature of poaching further destabilizes the population.

Which of the various deterministic and stochastic factors are important to consider in a PVA will depend on the species biology, the present population size and distribution, and the threats it faces. For example, orang utans may be threatened by forest destruction and other largely deterministic processes, but inbreeding and randomly skewed sex ratios resulting from highly stochastic processes are unlikely to be problems, at least not on a species-wide basis. On the other hand, even if the remnant Atlantic coastal rainforest of Brazil is secured for the future, the populations of golden lion tamarins (*Leontopithecus rosalia*) which can persist in that remnant forest are not sufficiently large to be stable in the face of stochastic threats (Seal et al. 1990; Rylands 1993/4; Ballou et al. 1997). The identification of the primary threats facing a taxon via a comprehensive PVA is important for conservation planning. For example, tamarin populations might be stabilized by the translocations and reintroductions that are underway and planned, but an orang utan PHVA recognized that releases of confiscated “pet” orang utans are unlikely to have a conservation benefit for those populations which are facing habitat destruction, not stochastic fluctuations and inbreeding. For many species, such as the whooping crane (*Grus americana*), the temporarily extinct-in-the-wild black-footed ferret (*Mustela nigripes*), and the Puerto Rican parrot (*Amazona vitatta*), only a single population persisted in the wild. Although those populations may have been maintained or even increased for a number of years, the principal threat was that a local catastrophe (e.g., disease epidemic, severe storm) could decimate the population (Clark 1989; Lacy et al. 1989; Mirande et al. 1991). The primary recovery actions therefore needed to include the establishment of additional populations. Tragically, some taxa, such as the eastern barred bandicoot (*Perameles gunnii*) in Australia, may be critically threatened simultaneously by deterministic factors and stochastic processes (Lacy and Clark 1990).

PVA is formally an assessment of the probability of extinction, but PVA methods often focus on other indicators of population health. Mean and variance in population growth (Lindenmayer and Lacy 1995a, 1995b, 1995c), changes in range, distribution, and habitat occupancy (Hanski and Gilpin 1991, 1997), and losses of genetic variability (Soulé et al. 1986; Lande and Barrowclough 1987; Seal 1992; Lacy and Lindenmayer 1995) can be analyzed and monitored. Although not yet common, monitoring of population health could also utilize measures of developmental stability (Clarke 1995), physiological parameters such as body condition (Altmann et al. 1993) or levels of the hormones related to stress and reproduction (Sapolsky 1982, 1986), or the stability of behavior and the social structure of the population (Samuels and Altmann 1991).

The interactions and synergism among threatening processes will often cause numerical, distributional, physiologic, behavioral, and genetic responses to concordantly reflect species decline and vulnerability. It remains important, however, to understand and target the primary causal factors in species vulnerability. The recent proposal to base IUCN categories of threat on quantified criteria of probability of extinction, or changes in such indicators as species range, numbers, and trends (Mace and Lande 1991; Mace et al. 1992; Mace and Stuart 1994; IUCN Species Survival Commission 1994) reflects the increased understanding of the extinction process that has accompanied the development of PVA, and

simultaneously demands that much more progress be made in developing predictive models, gathering relevant data on status and threats, and applying the PVA techniques.

Population and Habitat Viability Analysis (PHVA)

Population and Habitat Viability Analysis is a multi-faceted process or framework for assisting conservation planning, rather than a singular technique or tool. It is often interwoven with other techniques for managing complex systems, such as decision analysis (Maguire 1986; Maguire et al. 1990). Even when viewed as *the PHVA workshop*, all such conservation workshops involved and required substantial pre-workshop and post-workshop activities. Some PHVA workshops have been extended into multiple workshops and less formal, smaller collaborative meetings, often focused on subsets of the larger problems of species conservation.

Although PHVAs are diverse and not well defined, the PHVA process contains a number of critical components. First, it is essential to gather an array of experts who have knowledge of the species or problem. A PHVA is not required to bring together experts, but it often facilitates such sharing of expertise because the collective knowledge of many is essential for a useful PVA (in the narrow sense) to be completed. In addition to a diversity of people, a PHVA workshop also requires and therefore facilitates the involvement of a number of agencies and other concerned organizations. For example, the PVA on the two endemic primates of the Tana River Primate Reserve in Kenya (Seal et al. 1991) was convened by the Kenya Wildlife Service, facilitated by the IUCN SSC Captive Breeding Specialist Group, benefited from the expertise contributed by members of the IUCN SSC Primate Specialist Group, and was sponsored by the World Bank. The involvement of many agencies and interested parties is critical to endangered species recovery.

An early requirement, or prerequisite, of a PHVA workshop is to determine the conservation problem to be addressed, and to state the goals of the management plan. Many endangered species programs have not clearly identified their goals. For example, at a PHVA and Conservation Assessment and Management Plan workshop on the forest birds of the Hawaiian islands (Ellis et al. 1992a, 1992b), it became apparent that the agencies responsible for the conservation of Hawaii's bird fauna had not determined whether their goal was to prevent species extinctions, prevent taxa (species or subspecies) from becoming extirpated on any of the islands they presently inhabit, preserve species in sufficient numbers and distribution to allow them to continue to fill ecological roles in the biological communities, or the restoration of taxa to most or all parts of the original ranges. The management actions required to achieve these various levels of conservation are quite different. In contrast, a PHVA on the Grizzly Bear in the Central Rockies of Canada (Herrero and Seal 2000) clearly identified that provincial policy called for maintenance of stable or growing populations of the species. Thus, the criterion against which alternative management scenarios were judged was whether the PVA projections indicated that the populations would not decline.

PHVA workshops facilitate the assembly of all available data. Often, important information is found in the field notes of researchers or managers, in the heads of those who have worked with and thought about the problems of the species, and in unpublished agency reports, as well as in the published scientific literature. A pending PHVA can be the impetus that encourages the collection of data in anticipation of presentation, review, and analysis at the workshop. For example, a Sumatran Tiger PHVA helped stimulate the systematic collection of data on sightings and signs of tigers in protected areas throughout the island of Sumatra, and collation and integration with a Geographic Information System (GIS) map of habitats and human pressures on those habitats. The PHVA on the Grizzly Bear in the Central Canadian Rockies Ecosystem provided the opportunity for detailed habitat mapping data to be integrated with population biology data on the bears, resulting in the development of models which would allow projection of the impacts of habitat changes on the bear populations.

It is important to specify the assumptions that underlay a PHVA, and any consequent management recommendation. For example, the Hawaiian bird conservation efforts are constrained by a belief that no birds bred outside of the islands should ever be brought back to the islands for release. While this position derives from a reasonable concern for disease transmission (much of the decline of Hawaii's native birds is thought to be due to introduced avian diseases) as much as from any political or philosophical stand, any justification for the restriction must be questioned in light of the fact that wildlife agencies import and release, without quarantine, 1000s of exotic game birds onto the islands annually.

Once experts are assembled, problems stated and goals set, data gathered, and assumptions specified, then the PHVA process can proceed with what I describe as PVA in the narrow sense: estimation of the probability of population persistence. The available data are used to estimate the parameters that are needed for the model of population dynamics to be applied. Often, data are not available from which to estimate certain key parameters. In those cases, subjective and objective, but non-quantified, information might be solicited from the assembled experts, values might be obtained from data on related species, or a factor might simply be omitted from the model. While such a non-precise process might consist simply of intuitive judgments made by experts, it is important to specify how values for the parameters in the model were obtained. The resulting limitations of the analyses should be acknowledged, and a decision made if, how, by whom, and when the missing data would be collected so that more refined analyses could be conducted. With the PVA model, projections of the most likely fate, and distribution of possible fates, of the population under the specified assumptions are made.

Because so much of a PVA – the data, the model, and even the interpretation of output – is uncertain, a PVA that provides an estimate of the probability of extinction under a single scenario is of very limited usefulness. An essential component of the PHVA process, therefore, is sensitivity testing. Ranges of plausible values for uncertain parameters should be tested, to determine what effects those uncertainties might have on the results. In addition, several different PVA models might be examined at a PHVA workshop, or the same general model tested under different structural assumptions. Different participants in the process should assess and interpret the results. Such sensitivity testing reveals which components of the data, model, and interpretation have the largest impact on the population projections. This will indicate which aspects of the biology of the population and its situation contribute most to its vulnerability and, therefore, which aspects might be most effectively targeted for management. In addition, uncertain parameters that have a strong impact on results are those which might be the focus of future research efforts, to better specify the dynamics of the population. Close monitoring of such parameters might also be important for testing the assumptions behind the selected management options and for assessing the success of conservation efforts.

Closely parallel to the testing of uncertainties in the present situation is the testing of options for management. PVA modeling allows one to test the expected results of any given management action, under the assumptions of the model and within the limitations of present knowledge, on the computer before implementation in the field. This process can guide selection of the management options most likely, given current knowledge, to be effective, and will define target recovery goals that should be obtained if our knowledge is adequate and the recommended actions are followed. A PHVA workshop on the Black Rhinoceros in Kenya's 11 rhino sanctuaries (Foose et al. 1993) suggested that periodic movement of rhinos between fenced sanctuaries to reduce inbreeding and demographic fluctuations would be necessary to stabilize the populations in the smaller parks. Moreover, the modeling provided estimates of the rate at which the larger populations would be able to provide surplus animals for translocation.

It would be an error to assume that any PVA model incorporates everything of interest. A PVA simulation program can only include those processes that are known to the programmer. This will likely be a subset of what might be known to the field biologists, which in turn will definitely be a subset of

those processes that impact natural populations. A number of variables affecting population dynamics and viability are not yet commonly examined in PVA models. These include: social and ecological determinants of dispersal; complex social processes, such as the role of non-breeders in group stability and the impacts of other aspects of the social environment on reproductive success and survival; competitive, exploitative, or mutualistic interactions with other species experiencing their own population dynamics; and the effects of changes in the global environment. To date, most PVA models treat organisms as independent actors in spatially homogeneous physical, biotic, and social environments. There is tremendous opportunity and need for elaboration of PVA models, and it is likely that increasingly sophisticated models will also become more specific to the individual taxa and environments under study.

PHVA workshops must incorporate consideration of the assumptions of the PVA model used and the biases or limitations in interpretation that could result. PHVAs consider only those threatening processes of which we have knowledge, for which we can develop algorithms for modeling or other methods for analysis, and for which we have some data. As a result, it is likely that PVAs will underestimate the vulnerability of most populations to extinction, and that PHVA workshops will be less comprehensive than is desirable. We need always to be cognizant of the limits of our understanding of wildlife populations, and to include appropriate margins for error in our conservation strategies.

PVA is, by definition, an assessment of the probability of persistence of a population over a defined time frame. Yet, persistence of a population, while a necessary condition for effective conservation of natural systems, is often not sufficient. Prevention of extinction is the last stand of conservationists, but the goals should be higher: conservation of functional biological communities and ecosystems. PVA usually ignores the functional role of a species in a community, but a PHVA workshop should consider much more than the prevention of the final biological extinction of the taxon. A species, such as the American Bison (*Bison bison*), can be functionally extinct in terms of no longer filling its original role in nature, even as it is praised as a conservation success story and would be considered safe from extinction and viable.

The use of the PHVA process to help guide conservation decisions is not a singular event, in which an analysis can be completed, management actions recommended and implemented, and conservation thereby assured. The many uncertainties in the process mandate that PVA be used as a tool in an adaptive management framework, and a PHVA workshop is just one stage of an effective conservation strategy. In adaptive management, the lack of knowledge adequate to predict with certainty the best course of action is recognized, management actions are designed in such a way that monitoring will allow testing of the adequacy of our model and understanding, and corrective adjustments to management plans are made whenever the accumulating data suggest that the present course is inadequate to achieve the goals and that a better strategy exists (Holling 1978). The urgency of the biodiversity crisis will not permit us ethically to refrain from aggressive conservation action until we have scientifically sound understanding of all the factors that drive population, community, and ecosystem dynamics. PHVA provides a forum for making use of the information we do have, in a well-documented process that is open to challenge and improvement. PHVA workshops can, therefore, assist wildlife managers in the very difficult and important job of using science to safeguard the future of wildlife populations.

In summary, Population Viability Analysis (PVA) and Population and Habitat Viability Analysis (PHVA) refer to an array of interrelated and evolving techniques for assessing the survival probability of a population and possible conservation actions. It might be useful to restrict the term PVA to its original meaning -- the use of quantitative techniques to estimate the probability of population persistence under a chosen model of population dynamics, a specified set of biological and environmental parameters, and enumerated assumptions about human activities and impacts on the system. PHVA refers to a workshop approach to conservation planning, which elicits and encourages contributions from an array of experts

and stakeholders, uses PVA and other quantitative and non-quantitative techniques to assess possible conservation actions, and strives to achieve consensus on the best course of action from competing interests and perspectives, incomplete knowledge, and an uncertain future.

Many of the components of PVAs and PHVAs, even when used in isolation, can be effective educational and research tools. To be a useful framework for advancing the conservation of biodiversity, however, PHVA must incorporate all of: (1) collection of data on the biology of the taxon, status of its habitat, and threats to its persistence, (2) quantitative analysis of available data, (3) input of population status and identifiable threats to persistence into analytical or simulation models of the extinction process, (4) assessment of the probability of survival over specified periods of time, given the assumptions and limitations of the data and model used, (5) sensitivity testing of estimates of extinction probability across the range of plausible values of uncertain parameters, (6) specification of conservation goals for the population, (7) identification of options for management, (8) projection of the probability of population survival under alternative scenarios for future conservation action, (9) implementation of optimal actions for assuring accomplishment of conservation goals, (10) continued monitoring of the population, (11) reassessment of assumptions, data, models, and options, and (12) adjustment of conservation strategies to respond to the best information available at all times. There are many uncertain aspects of population dynamics, especially of endangered taxa, including few data on species biology and habitats, uncertain political and social climate for implementing conservation actions, and the unpredictability inherent in small populations due to the many stochastic forces that drive population dynamics.

The rapid development of PVA as a research and management tool, and the concurrent but not always parallel expansion of the scope of what conservation threats, options, and actions are considered in PHVA workshops, has led to confusion. Different people can describe rather distinct kinds of analyses with the same terminology, while others use different terms to describe nearly identical approaches. The ever-changing concepts of PVA and PHVA are confusing, but the flexibility of the processes is also their strength. Current tools are inadequate to address fully the challenges of stemming the losses of biodiversity. The PVA/PHVA framework allows and encourages rapid application of new tools, data, and interpretations into increasingly effective conservation programs.

Methods for Analyzing Population Viability

An understanding of the multiple, interacting forces that contribute to extinction vortices is a prerequisite for the study of extinction-recolonization dynamics in natural populations inhabiting patchy environments (Gilpin 1987), the management of small populations (Clark and Seebeck 1990), and the conservation of threatened wildlife (Shaffer 1981, 1990; Soulé 1987; Mace and Lande 1991).

Shaffer (1981) suggested several ways to conduct PVAs. Perhaps the most rigorous method, and the one that would produce the most defensible estimates, would be an empirical observation of the stability and long term fates of a number of populations of various sizes. Berger (1990) presented a good example of this approach, in which he observed that populations of bighorn sheep in the mountains of the western USA persisted only when the populations consisted of more than 100 animals. A few other studies of wildlife populations have provided empirical data on the relationship between population size and probability of extinction (e.g., Belovsky 1987; Thomas 1990), but presently only order of magnitude estimates can be provided for MVPs of vertebrates (Shaffer 1987). More empirical studies are needed, but the time and numbers of populations required for such studies are precluded in the cases of most species threatened with extinction -- exactly those for which estimates of population vulnerability are most urgently needed.

A more elegant and general approach to PVA is to develop analytical models of the extinction process that will allow calculation of the probability of extinction from a small number of measurable parameters.

Goodman's (1987) model of demographic fluctuations, and applications to conservation of the classic population genetic models of loss of genetic diversity by genetic drift (Franklin 1980; Soulé et al. 1986; Lande and Barrowclough 1987) are valuable efforts in this direction. Unfortunately, our understanding of population biology is not yet sufficient to provide fully adequate analytical models of the extinction process. For example, none of the existing analytical models incorporate all three of demographic, environmental, and genetic fluctuations, and thus they do not begin to model the array of extinction vortices described by Gilpin and Soulé (1986). Moreover, the analytical models make extremely simplifying assumptions about a number of the intricacies of population structure. For example, social groupings or preferences are often assumed to be invariant or lacking, resulting in random mating; and dispersal is usually assumed to be random between all sites (the "island model") or only to occur between adjacent sites (the "stepping stone model"). Much more work is needed either to develop more complex and flexible models or to demonstrate that the simple models are sufficient to provide guidance for conservation.

A third method of conducting a PVA is the use of computer simulation modeling to project the probability distribution of possible fates of a population. Simulation models can incorporate a very large number of threatening processes and their interactions, if the processes can be described in terms of quantitative algorithms and parameterized. Although many processes affecting small populations are intrinsically indeterminate, the average long-term fate of a population and the variance around the expectation can be studied with computer simulation models. The focus is on detailed and explicit modeling of the forces impinging on a given population, place, and time of interest, rather than on delineation of rules (which may not exist) that apply generally to most wildlife populations.

Modeling and Population Viability Analysis

A model is any simplified representation of a real system. We use models in all aspects of our lives, in order to: (1) extract the important trends from complex processes, (2) permit comparison among systems, (3) facilitate analysis of causes of processes acting on the system, and (4) make predictions about the future. A complete description of a natural system, if it were possible, would often decrease our understanding relative to that provided by a good model, because there is "noise" in the system that is extraneous to the processes we wish to understand. For example, the typical representation of the growth of a wildlife population by an annual percent growth rate is a simplified mathematical model of the much more complex changes in population size. Representing population growth as an annual percent change assumes constant exponential growth, ignoring the irregular fluctuations as individuals are born or immigrate, and die or emigrate. For many purposes, such a simplified model of population growth is very useful, because it captures the essential information we might need regarding the average change in population size, and it allows us to make predictions about the future size of the population. A detailed description of the exact changes in numbers of individuals, while a true description of the population, would often be of much less value because the essential pattern would be obscured, and it would be difficult or impossible to make predictions about the future population size.

In considerations of the vulnerability of a population to extinction, as is so often required for conservation planning and management, the simple model of population growth as a constant annual rate of change is inadequate for our needs. The fluctuations in population size that are omitted from the standard ecological models of population change can cause population extinction, and therefore are often the primary focus of concern. In order to understand and predict the vulnerability of a wildlife population to extinction, we need to use a model which incorporates the processes which cause fluctuations in the population, as well as those which control the long-term trends in population size. Many processes can cause fluctuations in population size: variation in the environment (such as weather, food supplies, and predation), genetic changes in the population (such as genetic drift, inbreeding, and response to natural selection), catastrophic effects (such as disease epidemics, floods, and droughts), decimation of the population or its

habitats by humans, the chance results of the probabilistic events in the lives of individuals (sex determination, location of mates, breeding success, survival), and interactions among these factors (Gilpin and Soulé 1986).

Models of population dynamics which incorporate causes of fluctuations in population size in order to predict probabilities of extinction, and to help identify the processes which contribute to a population's vulnerability, are used in Population Viability Analysis (PVA). For the purpose of predicting vulnerability to extinction, any and all population processes that impact population dynamics can be important. Much analysis of conservation issues is conducted by largely intuitive assessments by biologists with experience with the system. Assessments by experts can be quite valuable, and are often contrasted with "models" used to evaluate population vulnerability to extinction. Such a contrast is not valid, however, as *any* synthesis of facts and understanding of processes constitutes a model, even if it is a mental model within the mind of the expert and perhaps only vaguely specified to others (or even to the expert himself or herself).

A number of properties of the problem of assessing vulnerability of a population to extinction make it difficult to rely on mental or intuitive models. Numerous processes impact population dynamics, and many of the factors interact in complex ways. For example, increased fragmentation of habitat can make it more difficult to locate mates, can lead to greater mortality as individuals disperse greater distances across unsuitable habitat, and can lead to increased inbreeding which in turn can further reduce ability to attract mates and to survive. In addition, many of the processes impacting population dynamics are intrinsically probabilistic, with a random component. Sex determination, disease, predation, mate acquisition -- indeed, almost all events in the life of an individual -- are stochastic events, occurring with certain probabilities rather than with absolute certainty at any given time. The consequences of factors influencing population dynamics are often delayed for years or even generations. With a long-lived species, a population might persist for 20 to 40 years beyond the emergence of factors that ultimately cause extinction. Humans can synthesize mentally only a few factors at a time, most people have difficulty assessing probabilities intuitively, and it is difficult to consider delayed effects. Moreover, the data needed for models of population dynamics are often very uncertain. Optimal decision-making when data are uncertain is difficult, as it involves correct assessment of probabilities that the true values fall within certain ranges, adding yet another probabilistic or chance component to the evaluation of the situation.

The difficulty of incorporating multiple, interacting, probabilistic processes into a model that can utilize uncertain data has prevented (to date) development of analytical models (mathematical equations developed from theory) which encompass more than a small subset of the processes known to affect wildlife population dynamics. It is possible that the mental models of some biologists are sufficiently complex to predict accurately population vulnerabilities to extinction under a range of conditions, but it is not possible to assess objectively the precision of such intuitive assessments, and it is difficult to transfer that knowledge to others who need also to evaluate the situation. Computer simulation models have increasingly been used to assist in PVA. Although rarely as elegant as models framed in analytical equations, computer simulation models can be well suited for the complex task of evaluating risks of extinction. Simulation models can include as many factors that influence population dynamics as the modeler and the user of the model want to assess. Interactions between processes can be modeled, if the nature of those interactions can be specified. Probabilistic events can be easily simulated by computer programs, providing output that gives both the mean expected result and the range or distribution of possible outcomes. In theory, simulation programs can be used to build models of population dynamics that include all the knowledge of the system which is available to experts. In practice, the models will be simpler, because some factors are judged unlikely to be important, and because the persons who developed the model did not have access to the full array of expert knowledge.

Although computer simulation models can be complex and confusing, they are precisely defined and all the assumptions and algorithms can be examined. Therefore, the models are objective, testable, and open to challenge and improvement. PVA models allow use of all available data on the biology of the taxon, facilitate testing of the effects of unknown or uncertain data, and expedite the comparison of the likely results of various possible management options.

PVA models also have weaknesses and limitations. A model of the population dynamics does not define the goals for conservation planning. Goals, in terms of population growth, probability of persistence, number of extant populations, genetic diversity, or other measures of population performance must be defined by the management authorities before the results of population modeling can be used. Because the models incorporate many factors, the number of possibilities to test can seem endless, and it can be difficult to determine which of the factors that were analyzed are most important to the population dynamics. PVA models are necessarily incomplete. We can model only those factors which we understand and for which we can specify the parameters. Therefore, it is important to realize that the models probably underestimate the threats facing the population. Finally, the models are used to predict the long-term effects of the processes presently acting on the population. Many aspects of the situation could change radically within the time span that is modeled. Therefore, it is important to reassess the data and model results periodically, with changes made to the conservation programs as needed.

Dealing with uncertainty

It is important to recognize that uncertainty regarding the biological parameters of a population and its consequent fate occurs at several levels and for independent reasons. Uncertainty can occur because the parameters have never been measured on the population. Uncertainty can occur because limited field data have yielded estimates with potentially large sampling error. Uncertainty can occur because independent studies have generated discordant estimates. Uncertainty can occur because environmental conditions or population status have been changing over time, and field surveys were conducted during periods which may not be representative of long-term averages. Uncertainty can occur because the environment will change in the future, so that measurements made in the past may not accurately predict future conditions.

Sensitivity testing is necessary to determine the extent to which uncertainty in input parameters results in uncertainty regarding the future fate of the population. If alternative plausible parameter values result in divergent predictions for the population, then it is important to try to resolve the uncertainty with better data. Sensitivity of population dynamics to certain parameters also indicates that those parameters describe factors that could be critical determinants of population viability. Such factors are therefore good candidates for efficient management actions designed to ensure the persistence of the population.

The above kinds of uncertainty should be distinguished from several more sources of uncertainty about the future of the population. Even if long-term average demographic rates are known with precision, variation over time caused by fluctuating environmental conditions will cause uncertainty in the fate of the population at any given time in the future. Such environmental variation should be incorporated into the model used to assess population dynamics, and will generate a range of possible outcomes (perhaps represented as a mean and standard deviation) from the model. In addition, most biological processes are inherently stochastic, having a random component. The stochastic or probabilistic nature of survival, sex determination, transmission of genes, acquisition of mates, reproduction, and other processes preclude exact determination of the future state of a population. Such demographic stochasticity should also be incorporated into a population model, because such variability both increases our uncertainty about the future and can also change the expected or mean outcome relative to that which would result if there were no such variation. Finally, there is “uncertainty” which represents the alternative actions or interventions that might be pursued as a management strategy. The likely effectiveness of such management options

can be explored by testing alternative scenarios in the model of population dynamics, in much the same way that sensitivity testing is used to explore the effects of uncertain biological parameters.

Often, the uncertainty regarding a number of aspects of the population biology, current status, and threats to persistence is too large to allow scientifically accurate and reliable projections of population dynamics. Therefore, the predictions made from PVA models should be considered to be projections about what would most likely happen to the population *if* various hypotheses about the status of the populations and the threats are true. Conservation and management decisions must be made based on the most plausible hypotheses about the population status, before sufficient data could be collected to test those hypotheses scientifically. An important advantage of PVA models is that they forced systematic consideration and specification of the assumptions and hypotheses that must be made in the absence of adequate data. This facilitates careful reassessment and improvement in the analyses, as better data become available.

Questions that can be explored with PVA models

Below are some of the conservation and management questions that can be explored by Population Viability Analysis modeling. References describing uses of VORTEX give many examples of these and other applications of PVA techniques to guide conservation.

Using the best current information on the biology of the taxon and its habitat, are the populations projected to persist if conditions remain as they are now? Beyond just the persistence of the population, what is the most likely average population size, range of population sizes across years, and rate of loss of genetic variation? If the population is at risk of extinction, is the extinction expected to result primarily from negative average population growth (mean deaths exceeding mean births), from large fluctuations in numbers, from effects of accumulated inbreeding, or from a combination of these factors?

Given that there is considerable uncertainty about several aspects of the species biology and its habitat, is the population likely to persist across the plausible ranges of parameters that might characterize the population? In particular, how sensitive are the population dynamics to varying estimates of reproductive success, juvenile survival, adult survival, effects of natural catastrophes, initial population size, carrying capacity of the habitat, and dispersal among populations? Are there critical values for any of these parameters which demarcate a transition from a population that would be considered viable to one that is not?

Which factors have the greatest influence on the projected population performance? If important factors are identified, management actions might be designed to improve these factors or ameliorate the negative effects. How much change would be required in aspects of the population in order to ensure population survival?

What would be the effect of removing some individuals from the population? Would there be a significant benefit from supplementing the population with individuals translocated from other populations or released from captive breeding stocks? Can the population sustain controlled harvest? Can it sustain poaching?

Would a corridor connecting fragmented habitats improve long-term viability? Could the same effect be achieved by translocating a few individuals? What will happen to population viability if mortality increases for individuals dispersing between habitat patches?

What will happen to the wildlife population if trends in human populations and human impacts on the environment continue unabated?

The *VORTEX* Population Viability Analysis Model

The *VORTEX* computer program is a simulation of the effects of deterministic forces as well as demographic, environmental and genetic stochastic events on wildlife populations. It is an attempt to model many of the extinction vortices that can threaten persistence of small populations (hence, its name). *VORTEX* models population dynamics as discrete, sequential events that occur according to probabilities that are random variables following user-specified distributions. *VORTEX* simulates a population by stepping through a series of events that describe an annual cycle of a typical sexually reproducing, diploid organism: mate selection, reproduction, mortality, increment of age by one year, migration among populations, removals, supplementation, and then truncation (if necessary) to the carrying capacity. Although *VORTEX* simulates life events on an annual cycle, a user could model "years" that are other than 12 months duration. The simulation of the population is iterated many times to generate the distribution of fates that the population might experience.

VORTEX is an individual-based model. That is, it creates a representation of each animal in its memory and follows the fate of the animal through each year of its lifetime. *VORTEX* keeps track of the sex, age, and parentage of each animal. Demographic events (birth, sex determination, mating, dispersal, and death) are modeled by determining for each animal in each year of the simulation whether any of the events occur.

VORTEX requires a lot of population-specific data. For example, the user must specify the amount of annual variation in each demographic rate caused by fluctuations in the environment. In addition, the frequency of each type of catastrophe (drought, flood, epidemic disease) and the effects of the catastrophes on survival and reproduction must be specified. Rates of migration (dispersal) between each pair of local populations must be specified. Because *VORTEX* requires specification of many biological parameters, it is not necessarily a good model for the examination of population dynamics that would result from some generalized life history. It is most usefully applied to the analysis of a specific population in a specific environment.

In the program explanation that follows, demographic rates are described as constants specified by the user. Although this is the way the program is most commonly and easily used, *VORTEX* does provide the capability to specify most demographic rates as functions of time, density, and other parameters.

Demographic stochasticity

VORTEX models demographic stochasticity by determining the occurrence of probabilistic events such as reproduction, litter size, sex determination, and death with a pseudo-random number generator. For each life event, if the random value sampled from a specified distribution falls above the user-specified probability, the event is deemed to have occurred, thereby simulating a binomial process. Demographic stochasticity is therefore a consequence of the uncertainty regarding whether each demographic event occurs for any given animal.

Random deviates from binomial distributions, with mean p and standard deviation s , are obtained by first determining the integral number of binomial trials, N , that would produce the value of s closest to the specified value, according to:

$$N = \frac{p(1-p)}{s^2}$$

N binomial trials are then simulated by sampling from the uniform 0-1 distribution to obtain the desired result, the frequency or proportion of successes. If the value of N determined for a desired binomial

distribution is larger than 25, a normal approximation is used in place of the binomial distribution. This normal approximation must be truncated at 0 and at 1 to allow use in defining probabilities, although, with such large values of N , s is small relative to p and the truncation would be invoked only rarely. To avoid introducing bias with this truncation, the normal approximation to the binomial (when used) is truncated symmetrically around the mean.

Environmental variation

VORTEX can model annual fluctuations in birth and death rates and in carrying capacity as might result from environmental variation. To model environmental variation, each demographic parameter is assigned a distribution with a mean and standard deviation that is specified by the user. Annual fluctuations in probabilities of reproduction and mortality are modeled as binomial distributions. Environmental variation in carrying capacity is modeled as a normal distribution. Environmental variation in demographic rates can be correlated among populations.

Catastrophes

Catastrophes are modeled in *VORTEX* as random events that occur with specified probabilities. A catastrophe will occur if a randomly generated number between zero and one is less than the probability of occurrence. Following a catastrophic event, the chances of survival and successful breeding for that simulated year are multiplied by severity factors. For example, forest fires might occur once in 50 years, on average, killing 25% of animals, and reducing breeding by survivors 50% for the year. Such a catastrophe would be modeled as a random event with 0.02 probability of occurrence each year, and severity factors of 0.75 for survival and 0.50 for reproduction. Catastrophes can be local (impacting populations independently), or regional (affecting sets of populations simultaneously).

Genetic processes

VORTEX models loss of genetic variation in populations, by simulating the transmission of alleles from parents to offspring at a hypothetical neutral (non-selected) genetic locus. Each animal at the start of the simulation is assigned two unique alleles at the locus. Each offspring created during the simulation is randomly assigned one of the alleles from each parent. *VORTEX* monitors how many of the original alleles remain within the population, and the average heterozygosity and gene diversity (or “expected heterozygosity”) relative to the starting levels. *VORTEX* also monitors the inbreeding coefficients of each animal, and can reduce the juvenile survival of inbred animals to model the effects of inbreeding depression.

Inbreeding depression is modeled as a loss of viability of inbred animals during their first year. The severity of inbreeding depression is commonly measured by the number of “lethal equivalents” in a population (Morton et al. 1956). The number of lethal equivalents per diploid genome estimates the average number of lethal alleles per individual in the population if all deleterious effects of inbreeding were due entirely to recessive lethal alleles. A population in which inbreeding depression is one lethal equivalent per diploid genome may have one recessive lethal allele per individual, it may have two recessive alleles per individual, each of which confer a 50% decrease in survival, or it may have some other combination of recessive deleterious alleles which equate in effect with one lethal allele per individual.

VORTEX partitions the total effect of inbreeding (the total lethal equivalents) into an effect due to recessive lethal alleles and an effect due to loci at which there is heterozygote advantage (superior fitness of heterozygotes relative to all homozygote genotypes). To model the effects of lethal alleles, each founder starts with a unique recessive lethal allele (and a dominant non-lethal allele) at up to five modeled loci.

By virtue of the deaths of individuals that are homozygous for lethal alleles, such alleles can be removed slowly by natural selection during the generations of a simulation. This diminishes the probability that inbred individuals in subsequent generations will be homozygous for a lethal allele.

Heterozygote advantage is modeled by specifying that juvenile survival is related to inbreeding according to the logarithmic model:

$$\ln(S) = A - BF$$

in which S is survival, F is the inbreeding coefficient, A is the logarithm of survival in the absence of inbreeding, and B is the portion of the lethal equivalents per haploid genome that is due to heterozygote advantage rather than to recessive lethal alleles. Unlike the situation with fully recessive deleterious alleles, natural selection does not remove deleterious alleles at loci in which the heterozygote has higher fitness than both homozygotes, because all alleles are deleterious when homozygous and beneficial when present in heterozygous combination with other alleles. Thus, under heterozygote advantage, the impact of inbreeding on survival does not diminish during repeated generations of inbreeding.

Unfortunately, for relatively few species are data available to allow estimation of the effects of inbreeding, and the magnitude of these effects apparently varies considerably among species (Falconer 1981; Ralls et al. 1988; Lacy et al. 1993) and even among populations of the same species (Lacy et al. 1996). Even without detailed pedigree data from which to estimate the number of lethal equivalents in a population and the underlying nature of the genetic load (recessive alleles or heterozygote advantage), PVAs must make assumptions about the effects of inbreeding on the population being studied. If genetic effects are ignored, the PVA will overestimate the viability of small populations. In some cases, it might be considered appropriate to assume that an inadequately studied species would respond to inbreeding in accord with the median (3.14 lethal equivalents per diploid) reported in the survey by Ralls et al. (1988). In other cases, there might be reason to make more optimistic assumptions (perhaps the lower quartile, 0.90 lethal equivalents), or more pessimistic assumptions (perhaps the upper quartile, 5.62 lethal equivalents). In the few species in which inbreeding depression has been studied carefully, about half of the effects of inbreeding are due recessive lethal alleles and about half of the effects are due to heterozygote advantage or other genetic mechanisms that are not diminished by natural selection during generations of inbreeding, although the proportion of the total inbreeding effect can vary substantially among populations (Lacy and Ballou 1998).

A full explanation of the genetic mechanisms of inbreeding depression is beyond the scope of this manual, and interested readers are encouraged to refer to the references cited above.

VORTEX can model monogamous or polygamous mating systems. In a monogamous system, a relative scarcity of breeding males may limit reproduction by females. In polygamous or monogamous models, the user can specify the proportion of the adult males in the breeding pool. Males are randomly reassigned to the breeding pool each year of the simulation, and all males in the breeding pool have an equal chance of siring offspring.

Deterministic processes

VORTEX can incorporate several deterministic processes, in addition to mean age-specific birth and death rates. Density dependence in mortality is modeled by specifying a carrying capacity of the habitat. When the population size exceeds the carrying capacity, additional mortality is imposed across all age classes to bring the population back down to the carrying capacity. Each animal in the population has an equal probability of being removed by this truncation. The carrying capacity can be specified to change over time, to model losses or gains in the amount or quality of habitat.

Density dependence in reproduction is modeled by specifying the proportion of adult females breeding each year as a function of the population size. The default functional relationship between breeding and density allows entry of Allee effects (reduction in breeding at low density) and/or reduced breeding at high densities.

Populations can be supplemented or harvested for any number of years in each simulation. Harvest may be culling or removal of animals for translocation to another (unmodeled) population. The numbers of additions and removals are specified according to the age and sex of animals.

Migration among populations

VORTEX can model up to 50 populations, with possibly distinct population parameters. Each pairwise migration rate is specified as the probability of an individual moving from one population to another. Migration among populations can be restricted to one sex and/or a limited age cohort. Emigration from a population can be restricted to occur only when the number of animals in the population exceeds a specified proportion of the carrying capacity. Dispersal mortality can be specified as a probability of death for any migrating animal, which is in addition to age-sex specific mortality. Because of between-population migration and managed supplementation, populations can be recolonized. *VORTEX* tracks the dynamics of local extinctions and recolonizations through the simulation.

Output

VORTEX outputs: (1) probability of extinction at specified intervals (e.g., every 10 years during a 100 year simulation), (2) median time to extinction, if the population went extinct in at least 50% of the simulations, (3) mean time to extinction of those simulated populations that became extinct, and (4) mean size of, and genetic variation within, extant populations.

Standard deviations across simulations and standard errors of the mean are reported for population size and the measures of genetic variation. Under the assumption that extinction of independently replicated populations is a binomial process, the standard error of the probability of extinction is reported by *VORTEX* as:

$$SE(p) = \sqrt{\frac{p(1-p)}{n}}$$

in which the frequency of extinction was p over n simulated populations. Demographic and genetic statistics are calculated and reported for each subpopulation and for the metapopulation.

Appendix 2.

Primary changes from Vortex 9 to Vortex 10

Input files

The *VORTEX* 10 project files will be saved in .xml format. The change to .xml files was made because they can be more forgiving of possible errors in the data file format, and in the future this will make it easier to open old projects in upgraded versions of *VORTEX*. Be careful, however, that after you save a project in the .xml format you will later want to open that .xml file and not some prior .vpj file from *VORTEX* 9.

VORTEX 10 can still read *VORTEX* 9 vpj project files. However, because the syntax of some variables used in functions has been changed (see below), some editing of input values may be necessary to get a *VORTEX* 9 project to run correctly in *VORTEX* 10. Files saved in *VORTEX* 10 format cannot be read subsequently by *VORTEX* 9.

After reading any VORTEX 9 vpj files into VORTEX 10, go through all input sections to be certain that values were translated correctly between versions, and correct any that were not.

Although it is easiest to enter input values from the user interface, the project files that specify input values can be edited directly in a browser or text editor. However, editing the xml project files within a text editor is not recommended, because it is easy to accidentally change the file format in a way that makes it unreadable by *VORTEX*.

Population-based modeling

The simulation can be run as a population-based model, rather than as an individual-based model. In a population-based simulation, all genetic options and modeling are disabled, as is individual variation (demographic stochasticity). Population-based models will run *much* faster than do individual-based models.

VORTEX 10 has no hard-coded limit on population size. Computer RAM might limit population size if inbreeding depression is modeled. As with *VORTEX* 9, inbreeding calculations will be very slow if $N > 10,000$. If N typically stays above about 1000 throughout the simulation, it might be more efficient (much faster, with no detectable change in results) to run the scenario as a population-based simulation. Note that inbreeding calculations are fast (and do not require much memory) if inbreeding depression is modeled as due 100% to recessive lethal alleles (and if no function uses inbreeding, ‘I’, or kinship, ‘KIN’, as a variable).

Some new options

New Options, available on the Project Settings page, on standard input pages, or as undocumented options:

- Create an output a file of gene diversity (heterozygosity) each year of each iteration. (Note that an output file of N for each year of each iteration is now always produced.)
- Output a file of the value of PS1 through PSx (with the ‘x’ being a number) for each population, each year of each iteration.
- Output a file of the value of GS1 through GSx (with the ‘x’ being a number) each year of each iteration.
- Do not show messages to the user while running. (This can be useful if many scenarios are running when the computer will not be attended.)
- Delay 1st year mortality until Mortality is imposed on all age classes. Normally, *VORTEX* imposes the 1st year mortality immediately after a brood is produced. This avoids having the population grow to very large size during the Breed step when a species has high fecundity and high 1st year mortality. This immediate imposition of 1st year mortality allows these early deaths to be removed before the kinship calculations are done, often thereby allowing *VORTEX* to run much more quickly because large numbers of kinships don’t need to be calculated for newborns that immediately die. The option to delay 1st year mortality is provided for cases in which you want to tally, disperse, harvest, or otherwise consider newborns before they are subjected to the 1st year mortality. Note, however, that deaths due to inbreeding depression are imposed at birth, regardless of this option.
- During harvest, instead of killing individuals that are “harvested”, set their ISx variable to 0. If ‘x’ is not specified, it is assumed to be IS1. This option can be used, for example, to contracept a specific number of animals each year (assuming that you also set the probability of breeding to be a function of IS1).
- Obtain the mate for each female from an ISvar labeled “MATE”. If MATE = -1, then a mate is selected at random from the available males. The ISvar labeled “MATE” will be the index of the individual to be used as the mate (if available).
- Obtain mates from ISvar MATE (as with the above option), but look for the male with ISvar = “name”, rather than ISvar = “index” (sequence in the list).
- Read ISvars from the studbook file specifying the starting population.
- Make offspring dependent on the dam for x years. Thus, a newborn is specified to be dependent on the dam until it becomes x years old. This will use IS1as a place to store for each female the number of currently living, still dependent offspring. This means that you can specify, for example, that only females with no dependent young can breed each year. Moreover, if the dam dies, then all currently dependent offspring are killed.

Annual sequence

The sequence of events in the annual cycle can be specified to be something other than the default (EV [setting annual rates]; Breeding; Mortality; Aging; Dispersal; Harvest; Supplementation; Calculate Growth; Carrying Capacity; Update GSVars, PSVars, and ISVars; Census). Steps can be placed in any order, and steps can be repeated (e.g., there can be dispersal both before and after harvest). Updating of state variables can be done as one operation (with the order being GS, then PS, then IS), or the three levels of state variables can be updated independently.

Age classes for which input values need to be specified will be determined by which steps are placed before Aging. For example, if Harvest occurs before Aging in each year, then the 0 age class animals can be harvested. If two breeding cycles occur without an Aging step intervening, then 0 age class animals can breed.

State variables

Any number of state variables can be created.

GSvars, PSvars, and ISvars can be referenced in functions by either the state variable number (i.e., GS1, PS3, IS7), as before, or can be referenced by their labels. Be sure not to use a label that is the same as a built-in function (such as DAM, SIRE, PARITY), with the exception of a special use of MATE (see below).

Dispersal

New options in the Dispersal input section include:

Prevent individuals from dispersing into saturated populations. This option will stop individuals from entering a population if it is already at its carrying capacity, thereby preventing that immigration from immediately causing the population to exceed K and be truncated. If a dispersing individual had been scheduled to be moved into a population that is already at K, then the individual will move to a different population instead.

During dispersal, use the dispersal rates to specify a fixed N of individuals that will disperse, rather than a percent.

Reproductive System

Maximum ages of breeding for males and for females can be set to be less than the maximum age. This allows modeling of species that can have post-reproductive life spans.

Specified Initial Age Structure

An option was added so that the initial age structure of the population can be specified by a proportional distribution with a total initial N. *VORTEX* will then allocate the initial N for you according to the specified distribution.

Carrying Capacity

K can be specified to be a criterion other than a limiting N. E.g., K can be a limiting number of females, or of adults, or of individuals with IS1=1. The proportion by which K exceeds the limiting value will determine the proportion of individuals (across all age and sex classes) that will be removed.

Genetic Management

Genetic management options can be applied to all populations or to any subset of populations. Different options can be specified for each population.

The maternally inherited mtDNA haplotype is now modeled.

Mutation rates can be specified for modeled loci. The mutation rate (per individual per allele) is specified. The user can also specify which loci are subjected to mutation. If no allele frequency file has been provided, an infinite allele model of mutation is used. If an allele frequency file is provided, then a finite allele mutation model is used, with the mutation being sampled at random from the other alleles at the locus, as specified for the “supplementation”, or last, population.

A new option is available to allow specification of initial inbreeding coefficients and pairwise kinships among founders to be set to a value greater than 0. With this option, you can start with a population that is already inbred and low in gene diversity. This will cause inbreeding depression to be imposed immediately on their offspring (if inbreeding depression is chosen, and if not all of the inbreeding depression is due to recessive lethal alleles). Note that the higher inbreeding that occurs when this option is chosen will not affect the model of inbreeding depression due to recessive lethal alleles. Note also that this option will also not affect the mean expected and observed heterozygosity, or any calculations on the population mean inbreeding (which is always set at the mean homozygosity of modeled loci), but it will affect the inbreeding of individuals. The mean inbreeding can be determined by setting IS1 = I, and then setting PS1 = IMEAN1.

A new option is available for “Criteria for separating a long-term pair”. A function or number entered for this option will be used as the probability each year that a long-term pair will be separated. For example, if a function evaluates to 0.25, then with 25% probability the pair will be separated and new mates assigned.

Section Notes

Notes about input values are now entered within a text box for each input section. These notes are inserted into the file (with .inp extension) that lists all the input values for a scenario.

Output files

All output files created by *VORTEX* 9 are still available in *VORTEX* 10. However, there have been some minor changes to the .dat files that are used for graphing. Sometimes, but not always, you can open data that were created in *VORTEX* 10 for graphing in *VORTEX* 9. (It is better to stick with *VORTEX* 10, once a project has been run in the new version.)

VORTEX 10 places all output files (and a copy of the input file) into a subfolder labeled “VOutput”.

New standard output file:

*.N (a tally by year and iteration of the population size; the data that are used to create the run-time display of N)

New optional output files: See Special Options, above.

Sensitivity Tests

Sensitivity Tests are handled in a different way in *VORTEX* 10, so that they are both more flexible and powerful but also easier to use. Therefore, any STs defined in a *VORTEX* 9 project file will be ignored when the old vpj file is loaded into *VORTEX* 10. See a separate document describing the *VORTEX* 10 Sensitivity Tests for more information about how to use STs in *VORTEX* 10.

Text Output contains a tab to display a summary of results from Sensitivity Tests. Graphs of ST results include the same graphs available for all scenarios, and some additional plots that are useful for comparing the relative effects of variables tested in a ST. All the data that are used for ST graphs are available in files with extension .stdat (with the same data format as the .dat files used for graphing non-ST scenarios) – so the ST results can also be analyzed and graphed in other programs.

Tables and Graphs

This section now provides quick access to 6 standard graphs of simulation results, as well as the option for Custom Graphs that provides the same array of variables for graphing that were available in version 9.

Some of the arrangements of graphs that were available in *VORTEX* 9 – such as graphing Populations or Variables across the x-axis – are no longer available. (Often, those graphs were difficult to interpret and not very useful, and very few people ever used them.)

The populations to be graphed for the selected scenarios are now specified via a simple list of checkboxes. All years up to an optional user-specified limit are now shown on the graphs and tables, so there is no longer an option to plot only specific years from within the range.

Lines labels (as shown in the graph legend and on the table) can now be edited from a table that is accessed from an “Edit Line Labels” button below the graph. After editing, you must either Accept or Cancel your changes.

Functions

The variable for inbreeding (*I*) is expressed as a proportion (0 to 1 scale), rather than a percent.

Default values for any variables used in functions can be specified in the input on the State Variables page. If specified, these values will be used when evaluating functions at the start of a simulation for initial determination of deterministic growth rates and stable age distribution. For any variables pertaining to individuals (e.g., age, sex, genotypes), these default values will be each year when evaluating functions that are needed for population-level calculations, such as the number of pairings to reach *K* (if that option is chosen) or functions defining harvest or supplementation.

Order of precedence of operators, within any parentheses, follows standard rules, rather than the strict left to right application of operators of *VORTEX 9*.

Syntax for using GSvars, PSvars, ISvars, and related functions has changed, as follows:

GS1 [rather than GS(1)]
PS1 [rather than PS(1)]
PPS1(p) [rather than PPS(1;p)]
IS1 [rather than IS(1)]
ITOT1 [rather than ITOT(1)]
ISUM1 [rather than ISUM(1)]
IMEAN1 [rather than IMEAN(1)]
IMAX1 [rather than IMAX(1)]
IMIN1 [rather than IMIN(1)]

A number of new built-in variables and functions are available for use. See the manual section on Functions for a complete list. No longer available as built-in variables (but available via PSvars set to these values): FF, MM, JJ, UU, WW, XX. NN is still available, but now with syntax NNx (e.g., NN1, rather than NN(1))

Quick view of Deterministic Rates

When entering input values, hitting ctrl-D or clicking on the “Det.” icon will pop up a window that gives the deterministic rates (r, lambda, R0, and generation time) for each of the populations of the current scenario. This is a quick way to check to see if your input values will produce a positive population growth, in the absence of any stochastic variation.

Literature Cited

(still to be edited and updated for version 10)

- Akçakaya, H.R. 1997. *RAMAS Metapop: Viability Analysis for Stage-Structured Metapopulations (Version 2.0)*. Setauket, NY: Applied Biomathematics.
- Altmann, J., D. Schoeller, S.A. Altmann, P. Muruthi and R.M. Sapolsky. 1993. Body size and fatness of free-living baboons reflect food availability and activity levels. *American Journal of Primatology* 30:149-161.
- Alvarez, K. 1993. *Twilight of the Panther. Biology, Bureaucracy, and Failure in an Endangered Species Program*. Myakka River Publ., Sarasota, Florida.
- Ballou, J.D. 1983. Calculating inbreeding coefficients from pedigrees. Pages 509-520 in: Schonewald-Cox, C.M., S.M. Chambers, B. MacBryde, and W.L. Thomas (eds.). *Genetics and Conservation: A Reference for Managing Wild Animal and Plant Populations*. Menlo Park, California: Benjamin/Cummings.
- Ballou, J.D. 1997. Ancestral inbreeding only minimally affects inbreeding depression in mammalian populations. *Journal of Heredity* 88:169-178.
- Ballou, J.D. and R.C. Lacy. 1995. Identifying genetically important individuals for management of genetic diversity in pedigree populations. Pages 76-111 in J.D. Ballou, M. Gilpin, and T.J. Foose (eds.), *Population Management for Survival & Recovery. Analytical Methods and Strategies in Small Population Conservation*. New York: Columbia University Press.
- Ballou, J.D., R.C. Lacy, D. Kleiman, A. Rylands, and S. Ellis (eds.). 1997. *Leontopithecus II. The Second Population and Habitat Viability Assessment for Lion Tamarins (Leontopithecus)*. Apple Valley, MN: Conservation Breeding Specialist Group (SSC/IUCN).
- Belovsky, G.E. 1987. Extinction models and mammalian persistence. Pages 35-57 in: Soulé, M.E. (ed.). *Viable Populations for Conservation*. Cambridge: Cambridge University Press.
- Berger, J. 1990. Persistence of different-sized populations: an empirical assessment of rapid extinctions in bighorn sheep. *Conservation Biology* 4:91-98.
- Bonaccorso, F., P. Clark, P.S. Miller, and O. Byers (eds.). 1999. *Conservation Assessment and Management Plan for the Tree Kangaroos of Papua New Guinea and Population and Habitat Viability Assessment for Matschie's Tree Kangaroo: Final Report*. Apple Valley, MN: Conservation Breeding Specialist Group (SSC/IUCN).
- Bormann, F.H. and S.R. Kellert. 1991. *Ecology, Economics, and Ethics. The Broken Circle*. New Haven: Yale University Press.
- Boyce, M.S. 1992. Population viability analysis. *Annual Review of Ecology and Systematics* 23:481-506.
- Brussard, P. 1985. Minimum viable populations: how many are too few? *Restoration and Management Notes* 3:21-25.
- Burgman, M., S. Ferson and H.R. Akçakaya. 1993. *Risk Assessment in Conservation Biology*. New York: Chapman and Hall.
- Caswell, H. 2001. *Matrix Population Models*. 2nd ed. Sunderland, MA: Sinauer.
- Caughley, G. 1977. *Analysis of Vertebrate Populations*. London: John Wiley and Sons.

- Charlesworth, D., and B. Charlesworth. 1987. Inbreeding depression and its evolutionary consequences. *Annual Reviews of Ecology and Systematics* 18:237-268.
- Clark, T.W. 1989. *Conservation Biology of the Black-Footed Ferret*. Philadelphia: Wildlife Preservation Trust International.
- Clark, T.W. 1993. Creating and using knowledge for species and ecosystem conservation: Science, organizations, and policy. *Perspectives in Biology and Medicine* 36:497-525.
- Clark, T.W., R.M. Warneke and G.G. George. 1990. Management and conservation of small populations. Pages 1-18 in: Clark, T.W. and J.H. Seebeck (eds.). *Management and Conservation of Small Populations*. Brookfield, Illinois: Chicago Zoological Society.
- Clark, T.W., G.N. Backhouse and R.C. Lacy. 1991. The population viability assessment workshop: A tool for threatened species management. *Endangered Species Update* 8:1-5.
- Clark, T.W., and Seebeck, J.H. (eds.) 1990. *Management and Conservation of Small Populations*. Brookfield, Illinois: Chicago Zoological Society.
- Clarke, G.M. 1995. Relationships between developmental stability and fitness: Application for conservation biology. *Conservation Biology* 9:18-24.
- Crow, J.F. and Kimura, M. 1970. *Introduction to Population Genetics Theory*. New York: Harper and Row.
- Doughty, R.W. 1989. *Return of the Whooping Crane*. Austin : University of Texas Press.
- Edroma, E.L., N. Rosen, and P.S. Miller (eds.). 1997. *Conserving the Chimpanzees of Uganda: Population and Habitat Viability Assessment for Pan troglodytes schweinfurthii*. Apple Valley, MN: Conservation Breeding Specialist Group (SSC/IUCN).
- Ellis, S., K. Hughes, C. Kuehler, R.C. Lacy and U.S. Seal (eds.). 1992a. *Alala, Akohekohe, and Palila Population and Habitat Viability Assessment Reports*. Apple Valley, MN: Captive Breeding Specialist Group (SSC/IUCN).
- Ellis, S., C. Kuehler, R.C. Lacy, K. Hughes and U.S. Seal (eds.). 1992b. *Hawai`ian Forest Birds Conservation Assessment and Management Plan*. Apple Valley, MN: Captive Breeding Specialist Group (SSC/IUCN).
- Falconer, D.S. 1981. *Introduction to Quantitative Genetics*. 2nd ed. New York: Longman.
- Fisher, R.A. 1958. *The Genetical Theory of Natural Selection*. 2nd ed. New York; Dover.
- Fowler, C.W. 1981. Density dependence as related to life history strategy. *Ecology* 62:602-610.
- Franklin, I.R. 1980. Evolutionary change in small populations. Pages 135-149 in: Soulé, M.E. and B.A. Wilcox (eds.). *Conservation Biology: An Ecological/Evolutionary Perspective*. Sunderland, MA: Sinauer Associates.
- Foose, T.J., R.C. Lacy, R. Brett and U.S. Seal (eds.). 1993. *Kenyan Black Rhino Metapopulation Workshop Report*. Apple Valley, MN: Captive Breeding Specialist Group (SSC/IUCN).
- Gilpin, M.E. 1987. Spatial structure and population vulnerability. Pages 125-139 in: Soulé, M.E. (ed.). *Viable Populations for Conservation*. Cambridge: Cambridge University Press.
- Gilpin, M.E. 1989. Population viability analysis. *Endangered Species Update* 6:15-18.
- Gilpin, M.E., and M.E. Soulé. 1986. Minimum viable populations: processes of extinction. Pages 19-34 in: Soulé, M.E. (ed.). *Conservation Biology: The Science of Scarcity and Diversity*. Sunderland, MA: Sinauer Associates.

- Goodman, D. 1987. The demography of chance extinction. Pages 11 – 34 in: Soulé, M.E. (ed.). *Viable Populations for Conservation*. Cambridge: Cambridge University Press.
- Gunn, A., U.S. Seal, and P.S. Miller (eds.). 1998. *Population and Habitat Viability Assessment Workshop for Peary Caribou and Arctic-Island Caribou* (*Rangifer tarandus*). Apple Valley, MN: Conservation Breeding Specialist Group (SSC/IUCN).
- Hanski, I.A and M.E Gilpin. 1991. Metapopulation dynamics: A brief history and conceptual domain. *Biological Journal of the Linnean Society* 42:3-16.
- Hanski, I.A and M.E Gilpin (eds.). 1997. *Metapopulation Biology: Ecology, Genetics, and Evolution*. London: Academic Press.
- Hedrick, P.W. 1994. Purgling inbreeding depression and the probability of extinction: full-sib mating. *Heredity* 73:363-372.
- Hedrick, P.W. 2000. Genetics of Populations. 2nd Ed. Jones & Barlett Publishers, Sudbury, Massachusetts.
- Herrero, S., P.S. Miller and U. S. Seal (eds.). 2000. *Population and Habitat Viability Assessment Workshop for the Grizzly Bear of the Central Rockies Ecosystem* (*Ursus arctos horribilis*). Apple Valley, MN: Conservation Breeding Specialist Group (SSC/IUCN).
- Hobbs, N.T., and D.M. Swift. 1985. Estimates of habitat carrying capacity incorporating explicit nutritional constraints. *Journal of Wildlife Management* 49:814-822.
- Holling, C.S. (ed.). 1978. *Adaptive Environmental Assessment and Management*. International Series on Applied Systems Analysis 3, International Institute for applied systems analysis. Toronto: John Wiley and Sons.
- Hosack, D.A. 1998. *Population Viability Analysis Workshop for the Endangered Sonoran Pronghorn* (*Antilocapra americana sonoriensis*) *in the United States*. Washington, DC: Defenders of Wildlife.
- Ims, R.A., and N.G. Yoccoz. 1997. Studying transfer processes in metapopulations: Emigration, migration, and colonization. Pages 247 – 265 in: Hanski, I.A. and M.E. Gilpin (eds.). *Metapopulation Biology: Ecology, Genetics, and Evolution*. London: Academic Press.
- IUCN Species Survival Commission. 1994. *IUCN Red List Categories*. IUCN, Gland, Switzerland.
- Kjos, C., O. Byers, P.S. Miller, J. Borovansky, and U.S. Seal (eds.). 1998. *Population and Habitat Viability Assessment Workshop for the Winged Mapleleaf Mussel* (*Quadrula fragosa*): *Final Report*. Apple Valley, MN: Conservation Breeding Specialist Group (SSC/IUCN).
- Krebs, C.J. 1994. *Ecology: The Experimental Analysis of Distribution and Abundance*. 4th ed. New York: Harper Collins.
- Lacy, R.C. 1993a. VORTEX: A computer simulation model for Population Viability Analysis. *Wildlife Research* 20:45-65.
- Lacy, R.C. 1993b. Impacts of inbreeding in natural and captive populations of vertebrates: Implications for conservation. *Perspectives in Biology and Medicine*. 36:480-496.
- Lacy, R.C. 1993/1994. What is Population (and Habitat) Viability Analysis? *Primate Conservation* 14/15:27-33.
- Lacy, R.C. 1997. Importance of genetic variation to the viability of mammalian populations. *Journal of Mammalogy* 78:320-335.
- Lacy, R.C., G. Alaks, and A. Walsh. 1996. Hierarchical analysis of inbreeding depression in *Peromyscus polionotus*. *Evolution* 50:2187-2200.

- Lacy, R.C. and J.D. Ballou. 1998. Effectiveness of selection in reducing the genetic load in populations of *Peromyscus polionotus* during generations of inbreeding. *Evolution* 52:900-909.
- Lacy, R.C. and T.W. Clark. 1990. Population viability assessment of the eastern barred bandicoot in Victoria. Pages 131-146 in: Clark, T.W. and J.H. Seebek (eds.). *Management and Conservation of Small Populations*. Brookfield, IL: Chicago Zoological Society.
- Lacy, R.C., N.R. Flesness and U.S. Seal (eds.). 1989. *Puerto Rican Parrot Population Viability Analysis. Report to the U.S. Fish and Wildlife Service*. Apple Valley, MN: Captive Breeding Specialist Group (SSC/IUCN).
- Lacy, R.C. and D.B. Lindenmayer. 1995. A simulation study of the impacts of population sub-division on the mountain brushtail possum, *Trichosurus caninus* Ogilby (Phalangeridae: Marsupialia), in south-eastern Australia. II. Loss of genetic variation within and between sub-populations. *Biological Conservation* 73:131-142.
- Lacy, R.C., and P.S. Miller. 2002. Incorporating human populations and activities into population viability analysis. Pages 490-510 in S.R. Beissinger and D.R. McCullough (eds.), *Population Viability Analysis*. Chicago: University of Chicago Press.
- Lacy, R.C., Petric, A.M., and Warneke, M. 1993. Inbreeding and outbreeding depression in captive populations of wild species. Pages 352-374 in Thornhill, N.W. (ed.). *The Natural History of Inbreeding and Outbreeding*. Chicago: University of Chicago Press.
- Lande, R. and G.F. Barrowclough. 1987. Effective population size, genetic variation, and their use in population management. Pages 87-123 in Soulé, M.E. (ed.). *Viable Populations for Conservation*. Cambridge: Cambridge University Press.
- Latour, A. (1986). Polar normal distribution. *Byte* (August 1986):131-2.
- Lindenmayer, D.B. and R.C. Lacy. 1995a. Metapopulation viability of Leadbeater's Possum, *Gymnobelideus leadbeateri*, in fragmented old-growth forests. *Ecological Applications* 5:164-182.
- Lindenmayer, D.B. and R.C. Lacy. 1995b. Metapopulation viability of arboreal marsupials in fragmented old-growth forests: Comparison among species. *Ecological Applications* 5:183-199.
- Lindenmayer, D.B. and R.C. Lacy. 1995c. A simulation study of the impacts of population sub-division on the mountain brushtail possum, *Trichosurus caninus* Ogilby (Phalangeridae: Marsupialia), in south-eastern Australia. I. Demographic stability and population persistence. *Biological Conservation* 73:119-129.
- Lindenmayer, D.B., T.W. Clark, R.C. Lacy and V.C. Thomas. 1993. Population viability analysis as a tool in wildlife conservation policy: A review with reference to Australia. *Environmental Management* 17:745-758.
- Mace, G.M. and R. Lande. 1991. Assessing extinction threats: Toward a re-evaluation of IUCN threatened species categories. *Conservation Biology* 5:148-157.
- Mace, G. et al. 1992. The development of new criteria for listing species on the IUCN Red List. *Species* 19:16-22.
- Mace, G. and S. Stuart. 1994. Draft IUCN Red List Categories, Version 2.2. *Species* 21-22:13-24.
- MacNab, J. 1985. Carrying capacity and related slippery shibboleths. *Wildlife Society Bulletin* 13:403-410.

- Maguire, L.A. 1986. Using decision analysis to manage endangered species populations. *Journal of Environmental Management* 22:345-360.
- Maguire, L.A., R.C. Lacy, R.J. Begg and T.W. Clark. 1990. An analysis of alternative strategies for recovering the eastern barred bandicoot in Victoria. Pages 147-164 in: Clark, T.W. and J.H. Seebeck (eds.). *Management and Conservation of Small Populations*. Brookfield, IL: Chicago Zoological Society.
- Manansang, J., A. MacDonald, D. Siswomartono, P.S. Miller, and U.S. Seal (eds.). 1996. *Population and Habitat Viability Assessment for the Babirusa* (*Babirousa babyrussa*). Apple Valley, MN: Conservation Breeding Specialist Group (SSC/IUCN).
- Miller, P.S., and R.C. Lacy. 2003a. Integrating the human dimension into endangered species risk assessment. Pages 41-63 in: F.R. Westley and P.S. Miller, eds. *Experiments in Consilience: Integrating Social and Scientific Responses to Save Endangered Species*. Island Press, Washington, DC.
- Miller, P.S., and R.C. Lacy. 2003b. Metamodels as a tool for risk assessment. Pages 333-351 in: F.R. Westley and P.S. Miller, eds. *Experiments in Consilience: Integrating Social and Scientific Responses to Save Endangered Species*. Island Press, Washington, DC.
- Mirande, C., R. Lacy, and U. Seal (eds.). 1991. *Whooping Crane (Grus americana) Conservation Viability Assessment Workshop Report*. Apple Valley, MN: Captive Breeding Specialist Group (SSC/IUCN).
- Morton, N.E., Crow, J.F., and Muller, H.J. 1956. An estimate of the mutational damage in man from data on consanguineous marriages. *Proceedings of the National Academy of Sciences, USA* 42:855-863.
- Nei, M. 1987. Molecular Evolutionary Genetics. Columbia University Press, NY.
- O'Brien, S.J. and Evermann, J.F. 1988. Interactive influence of infectious diseases and genetic diversity in natural populations. *Trends in Ecology and Evolution* 3:254-9.
- Odum, A., et al (eds.). 1993. *Aruba Island Rattlesnake Population and Habitat Viability Assessment (PHVA) Workshop*. Apple Valley, MN: Captive Breeding Specialist Group (SSC/IUCN).
- Pergams, O.R.W., R.C. Lacy, and M.V. Ashley. 2000. Conservation and management of Anacapa Island deer mice. *Conservation Biology* 14:819-832.
- Petit, S., and L. Pors. 1996. Survey of columnar cacti and carrying capacity for nectar-feeding bats on Curaçao. *Conservation Biology* 10:769-775.
- Pielou, E.C. 1977. *Mathematical Ecology*. New York: John Wiley and Sons.
- Ralls, K., Ballou, J.D., and Templeton, A.R. 1988. Estimates of lethal equivalents and the cost of inbreeding in mammals. *Conservation Biology* 2:185-93.
- Ricklefs, R.E. 1979. *Ecology*. 2nd ed. New York: Chiron.
- Robertson, A. (1960). A theory of limits in artificial selection. *Proceedings Royal Society of London* 153B:234-49.
- Rohlf, F.J., and R.R. Sokal. 1981. *Statistical Tables*. 2nd ed. New York: W.H. Freeman and Company.
- Ruggiero, L.F., G.D. Hayward and J.R. Squires. 1994. Viability analysis in biological evaluations: Concepts of population viability analysis, biological population, and ecological scale. *Conservation Biology* 8:364-372.
- Rylands, A.B. 1993/1994. Population viability analyses and the conservation of the lion tamarins, *Leontopithecus*, of south-east Brazil. *Primate Conservation* 14/15: 34-42.

- Samuels, A. and J. Altmann. 1991. Baboons of the Amboseli basin: Demographic stability and change. *International Journal of Primatology* 12:1-19.
- Sapolsky, R.M. 1982. The endocrine stress-response and social status in the wild baboon. *Hormones and Behavior* 15:279-292.
- Sapolsky, R.M. 1986. Endocrine and behavioral correlates of drought in the wild baboon. *American Journal of Primatology* 11:217-227.
- Seal, U.S. (ed.). 1992. *Genetic Management Strategies and Population Viability of the Florida Panther* (*Felis concolor coryi*). Apple Valley, MN: Captive Breeding Specialist Group (SSC/IUCN).
- Seal, U.S. (ed.). 1994. *Attwater's Prairie Chicken Population and Habitat Viability Assessment*. Apple Valley, MN: Captive Breeding Specialist Group (SSC/IUCN).
- Seal, U.S. and R.C. Lacy (eds.). 1989. *Florida Panther Population Viability Analysis. Report to the U.S. Fish and Wildlife Service*. Apple Valley, MN: Captive Breeding Specialist Group (SSC/IUCN).
- Seal, U.S., J.D. Ballou and C.V. Padua (eds.). 1990. *Leontopithecus Population Viability Analysis Workshop Report*. Apple Valley, MN: Captive Breeding Specialist Group (SSC/IUCN).
- Seal, U.S., R.C. Lacy, K. Medley, R. Seal and T.J. Foose (eds.). 1991. *Tana River Primate Reserve Conservation Assessment Workshop Report*. Apple Valley, MN: Captive Breeding Specialist Group (SSC/IUCN).
- Selander, R.K. 1983. Evolutionary consequences of inbreeding. Pages 201-215 in: Schonewald-Cox, C.M, S.M. Chambers, B. MacBryde, and W.L. Thomas (eds.). *Genetics and Conservation: A Reference for Managing Wild Animal and Plant Populations*. Menlo Park, CA: Benjamin/Cummings.
- Shaffer, M.L. 1981. Minimum population sizes for species conservation. *Bioscience* 1:131-134.
- Shaffer, M.L. 1987. Minimum viable populations: Coping with uncertainty. Pages 69-86 in: Soulé, M.E. (ed.). *Viable Populations for Conservation*. Cambridge: Cambridge University Press.
- Shaffer, M.L. 1990. Population viability analysis. *Conservation Biology* 4:39-40.
- Simberloff, D.A. 1986. The proximate causes of extinction. Pages 259-276 in: Raup, D.M., and D. Jablonski (eds.). *Patterns and Processes in the History of Life*. Berlin: Springer-Verlag.
- Simberloff, D.A. 1988. The contribution of population and community biology to conservation science. *Annual Review of Ecology and Systematics* 19:473-511.
- Simmons, M.J., and J.F. Crow. 1977. Mutations affecting fitness in *Drosophila* populations. *Annual Review of Genetics* 11:49-78.
- Sokal, R.R., and F.J. Rohlf. 1994. *Biometry*. 3rd ed. New York: W.H. Freeman and Company.
- Soulé, M.E. (ed). 1987. *Viable Populations for Conservation*. Cambridge: Cambridge University Press.
- Soulé, M., M. Gilpin, W. Conway and T. Foose. 1986. The millennium ark: How long a voyage, how many staterooms, how many passengers? *Zoo Biology* 5:101-113.
- Starfield, A.M. and A.L. Bleloch. 1986. *Building Models for Conservation and Wildlife Management*,. New York: Macmillan.
- Thomas, C.D. 1990. What do real population dynamics tell us about minimum population sizes? *Conservation Biology* 4:324-327.
- Walker, S. and S. Molur (eds.). 1994. *Population and Habitat Viability Analysis (PHVA) Workshop for Indian/Nepali Rhinoceros*. Zoo Outreach Organisation/CBSG India, Coimbatore.

- Werikhe, S., L. Macfie, N. Rosen, and P.S. Miller (eds.). 1998. *Can the Mountain Gorilla Survive? Population and Habitat Viability Assessment Workshop for Gorilla gorilla beringei*. Apple Valley, MN: Conservation Breeding Specialist Group (SSC/IUCN).
- Wright, S. 1977. *Evolution and the Genetics of Populations. Vol. 3. Experimental Results and Evolutionary Deductions*. Chicago: University of Chicago Press.
- Zar, J.H. 1996. *Biostatistical Analysis*. 3rd ed. Englewood Cliffs: Prentice Hall.

Reprints

Lacy, R.C. 2000. Structure of the VORTEX simulation model for population viability analysis. Ecological Bulletins 48:191-203.

(Reprinted with permission of the publisher.)

Note: This paper describes the structure of *VORTEX* version 9. It is still largely appropriate for version 10, but you should be aware that the program now has options that were not available in version 9 (see above).

Lacy, R.C. 2000. Considering threats to the viability of small populations using individual-based models. Ecological Bulletins 48:39-51.

(Reprinted with permission of the publisher.)

Structure of the VORTEX simulation model for population viability analysis

Robert C. Lacy

Lacy, R. C. 2000. Structure of the VORTEX simulation model for population viability analysis. – *Ecol. Bull.* 48: 191–203.

The structure of the VORTEX computer simulation model for population viability analysis is outlined. The program flow is described here in order to provide a detailed specification of the structure of a widely used population viability analysis model.

VORTEX is an individual-based simulation program that models the effects of mean demographic rates, demographic stochasticity, environmental variation in demographic rates, catastrophes, inbreeding depression, harvest and supplementation, and metapopulation structure on the viability of wildlife populations. The model facilitates analysis of density-dependent reproduction and changing habitat availability, and most demographic rates can optionally be specified as flexible functions of density, time, population gene diversity, inbreeding, age, and sex. VORTEX projects changes in population size, age and sex structure, and genetic variation, as well as estimating probabilities and times to extinction and recolonization.

R. C. Lacy (rlacy@ix.netcom.com), Dept of Conservation Biology, Daniel F. & Ada L. Rice Center, Chicago Zoological Society, Brookfield, IL 60513, USA.

“VORTEX and like programs do exactly what they are told to do, as constrained by the static single-species models that provide their structure. They can be useful for various purposes so long as the user understands what the programs are doing ...” (Caughley and Gunn 1996, p. 208).

The complexity and multiplicity of processes influencing the dynamics of natural populations of animals and plants means that population viability analysis (PVA) models are also frequently complex. Different models incorporate different population processes. Individual population processes can be modeled in various ways, requiring different sets of driving variables, using different equations to define the processes, and providing different output to describe the population dynamics. Users of PVA models should understand the basic structure of the models they use, and it is important that models used for scientific studies and conservation efforts can be examined and replicated. Yet often the details of PVA computer programs

are not available to the users, because the code is proprietary information or otherwise not provided to users, or simply because the task of reading and understanding the source code for large and complex programs is formidable.

One possible remedy to the problem of PVA users needing to understand the models being used is for practitioners to develop their own computer programs. This would result in the user having a full understanding of a model that would be specifically designed for the analysis. Development of user-specific and case-specific models is usually not practical, however, as many population biologists are not skilled computer programmers, and the time required to develop a complex model is often prohibitive. Moreover, a complex computer program developed by and used by one person will sometimes contain serious programming errors. The testing of programs that are widely used may be a necessary prerequisite for reliable population viability analyses to be employed effectively in biodi-

versity conservation. Finally, the flexibility and expansive capabilities of generic PVA software to model a large diversity of population processes will often lead PVA practitioners to consider threats to population viability that would otherwise have been neglected.

Widely available PVA software can serve the same role as do statistical analysis packages. The ease of use, flexible application to diverse needs, and extensive prior testing facilitate many applications that would not otherwise be attempted. Ideally, perhaps, all users of statistical methods would write their own programs or otherwise study the code of the software entrusted for the analyses. More practically, confidence is gained in the reliability of generic software tools as more people use the programs and compare the generated results to expectations from statistical theory and to results for simple and well known cases. Also, users of statistical software are expected to be sufficiently familiar with the methods of statistical analysis to be able to choose appropriate models to apply to their problem, to be able to provide the proper input, and to be able to interpret the results.

Unlike the situation for statistical methods, however, there are not yet widely accepted and published accounts of standard methods for population viability analysis. The methods of population-based models (e.g., Starfield and Bleloch 1986, Burgman et al. 1993) are extensions of the methods of population ecology and demography (e.g., Pielou 1977, Caswell 1989), but many details of model construction require decisions about algorithms and methods that are not fully delineated in general treatments. The methods of individual-based PVA models have been only cursorily described in the scientific literature. Below is an outline of one widely used PVA software package, VORTEX, ver. 8.20. The basic approach taken in the VORTEX model is described in Lacy (1993), in Lindenmayer et al. (2000) and other papers describing applications of VORTEX, and in the software manual (Miller and Lacy 1999). Detailed documentation of the program flow and algorithms is provided here, so that users of VORTEX can confirm that the model is performing the analyses that are intended, and so that PVA practitioners in general can see an example of the structure of an individual-based PVA model. The VORTEX program is available at <http://www2.netcom.com/-rlacy/vortex.html>.

The pseudo-code presented below is an English-language outline of the program flow and primary algorithms used by VORTEX (which is written in the C programming language). This pseudo-code omits coding for: 1) input routines for reading parameters from files and/or keyboard; 2) output routines for writing results to files; 3) specification of default parameter values; 4) checks for illegal values, error handling; 5) memory management and initialization of memory; 6) details of C coding to achieve algorithms; 7) routines for on-line help; 8) routines for graphical display of functions specifying demographic rates, population sizes during simulations, simulation re-

sults; 9) routines for evaluating equations that specify demographic rates (e.g., breeding, mortality) as functions of population and individual variables (e.g., population size, gene diversity, year, age, sex, inbreeding) (see note 3 below); 10) tallies of mean within-population statistics and metapopulation summaries; 11) algorithms for calculating basic statistics, such as means, standard deviations, standard errors, and medians across years and across iterations.

Variables for storing input, intermediate calculations, and output are indicated in the pseudo-code by italicized labels. Many of the variables are arrays (e.g., a value stored for each population, or for each age class, or for each individual), as suggested by the loops within which they are calculated and used. The indices of such arrays are indicated within brackets (e.g., *MortalityRate[p][s][x]*, for each population, *p*, sex, *s*, and age, *x*). VORTEX uses many more variables (not shown in the pseudo-code) for facilitating calculations and accumulating sums, sums of squares, and other components needed for the basic statistics reported in the output.

In the pseudo-code, loops are indicated with FOR: and END LOOP statements, or by WHILE: and END WHILE statements. Conditional actions are indicated by IF: and END IF statements, or by IF:, ELSE:, and END IF/ELSE statements. BREAK indicates that program flow exits from the bottom of a loop. CONTINUE indicates that program flow jumps back to the next value at the top of the loop. Multiplication is indicated by the asterisk (*) symbol; ^ indicates exponentiation; SQRT indicates the positive square root.

Function modules defined outside of the main body of the pseudo-code program are labeled in the form FUNCTION(), and are specified below the main VORTEX() program. The actual C code is subdivided into many smaller functions; the pseudo-code shows only the flow of the overall program and its largest modules. The functions RAND() and NRAND() indicate, respectively, that a random number is generated from the uniform 0-1 distribution or from a unit normal distribution.

Explanatory comments, following pseudo-code sections, are preceded by //. More extensive explanations are given in notes following the code.

As an individual-based PVA simulation model, VORTEX represents each individual in memory, simulates life events (such as sex determination, breeding, mortality, and dispersal) which could occur to each individual, and monitors the status of each individual and the population as a whole. The characteristics tracked for each animal are: sex, alive/dead status, population membership, age, inbreeding coefficient, and two alleles at each of six loci. In addition, VORTEX maintains a matrix of kinship coefficients between all pairs of living animals, as this provides inbreeding coefficients for any offspring.

VORTEX models changes to a population as a series of discrete events that occur once per year (or other time interval). The annual sequence of demographic events is:

breeding; mortality; age 1 yr; migrate (disperse) among populations; harvest (managed removals); supplementation (managed additions); carrying capacity truncation; census (Fig. 1). Occurrences of events are probabilistic; demographic stochasticity emerges from chance variation in which individuals breed, die, and are of each sex. Envi-

ronmental variation in demographic rates is imposed by sampling rates from specified distributions during each simulated year. Catastrophes, which occur with specified probabilities, cause one-year reductions in reproduction and survival. Genetic effects are modeled as reduced survivorship of inbred individuals.

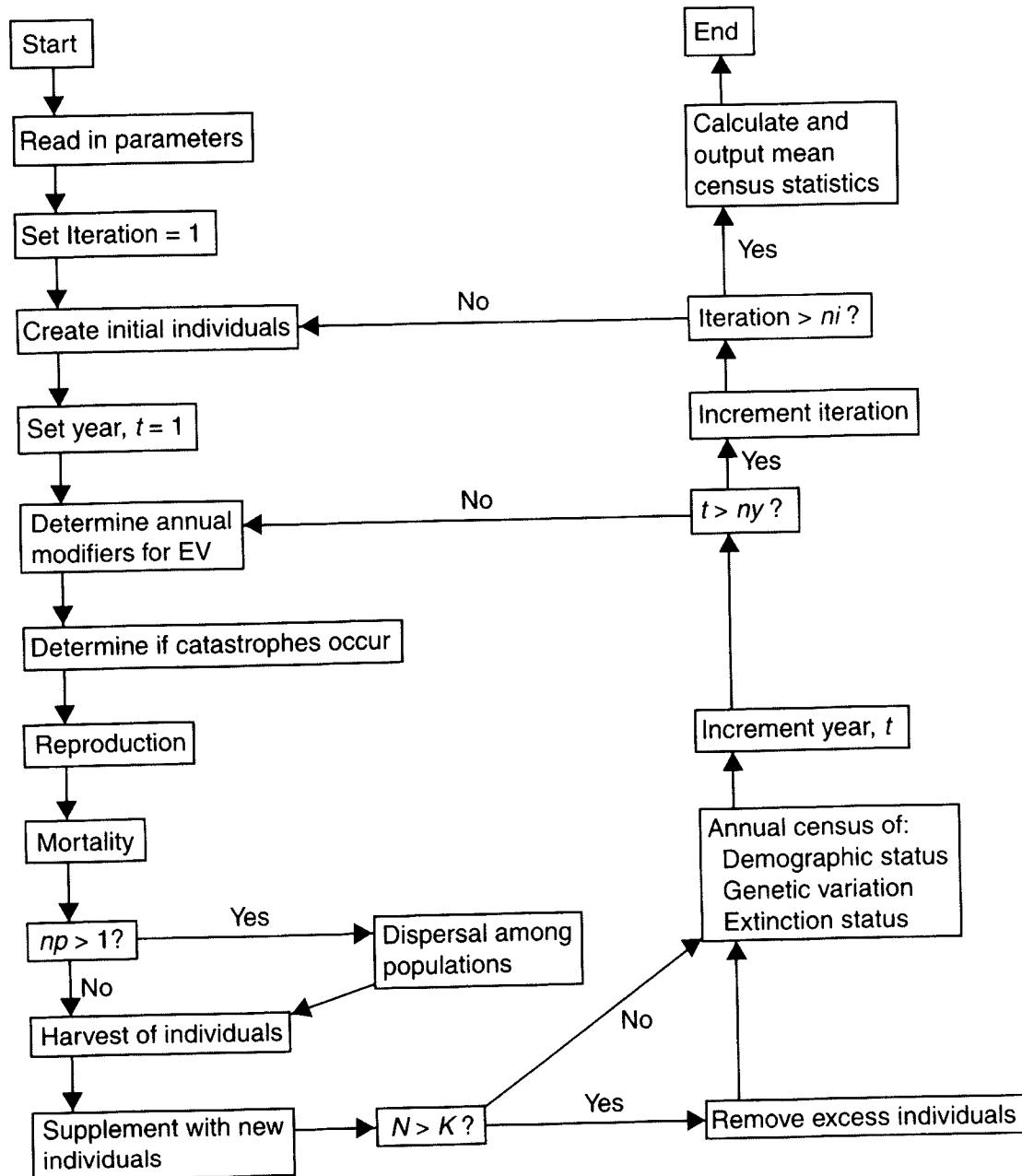


Fig. 1. Flow chart of the primary components of the Vortex simulation. Each step from "Create initial individuals" through "Annual census" is applied to each population in a modeled metapopulation. t = year; ny = number of years simulated; np = number of populations; ni = number of iterations; N = population size; K = carrying capacity; EV = environmental variation.

VORTEX program pseudo-code

```

BEGIN PROGRAM VORTEX():
Initialize random number generator; // See Note 1.
FOR (each scenario):
    READ_SPECIES_PARAMETERS();
    IF (NumberOfPopulations > 1):
        READ_MIGRATION_PARAMETERS();
// VORTEX describes dispersal between populations as
"migration."
    END IF
    FOR (each population, p):
        READ_POPULATION_PARAMETERS(p);
    END (population) LOOP
    FOR (each population, pSource):
// Calculate cumulative migration rates for each pairwise
transition between populations.
        Set CumMigrationProb[pSource][1] =
        MigrationProb[pSource][1];
        FOR (each destination population, pDestination,
greater than 1):
            Set CumMigrationProb[pSource][pDestination]
            = CumMigrationProb[pSource][pDestination
-1] + MigrationProb[pSource][pDestination];
        END (pDestination) LOOP
    END (pSource) LOOP
    Set NumberLethals = InbreedingGeneticLoad *
    ProportionLoadDueToLethals;
    Set LethalEquivalents = InbreedingGeneticLoad * (1 -
    ProportionLoadDueToLethals);
// See Note 2.
    FOR (each population, p):
        Set GlobalBreedEV[p] = BreedEV[p] *
        EVConcordanceAmongPopulations;
        Set LocalBreedEV[p] = SQRT( BreedEV[p] ^2 -
        GlobalBreedEV[p] ^2);
// Partition Environmental Variation in breeding
(BreedEV) into the component that is common to all
populations (GlobalBreedEV) and the component that is
specific to each population. (LocalBreedEV); TotalEV ^ 2
= GlobalEV ^ 2 + LocalEV ^ 2
// Note: EVs are given as standard deviations.
        FOR (each sex, s):
            FOR (each age, x, up to age of breeding):
                Set GlobalMortEV[p]/[s]/[x] =
                MortEV[p]/[s]/[x] *
                EVConcordanceAmongPopulations;
                Set LocalMortEV[p]/[s]/[x] =
                SQRT(MortEV[p]/[s]/[x] ^2 -
                GlobalMortEV[p]/[s]/[x] ^2);
// Partition EV in mortality (MortEV) into the compo-
nent that is common to all populations (GlobalMortEV)
and the component that is specific to each population
(LocalMortEV).
            END (age) LOOP
        END (sex) LOOP
Set GlobalKEV[p] = KEV[p] *
EVConcordanceAmongPopulations;
Set LocalKEV[p] = SQRT( KEV[p] ^2 -
GlobalKEV[p] ^2);
// Partition EV in carrying capacity (KEV) into the com-
ponent that is common to all populations (GlobalKEV)
and the component that is specific to each population
(GlobalKEV).
END (population) LOOP
FOR (each iteration):
    FOR (each population):
        Create initial individuals, assigning population,
        sex, age, alive/dead status, inbreeding coefficient
        and kinships (initially 0), and alleles at six loci;
        FOR (each of five non-neutral loci, l):
            FOR (each founder allele, a):
// The probability that a given founder allele is a lethal is
NumberLethals / 10, because there are 10 alleles across the
five diploid loci.
                IF (RAND() < NumberLethals / 10):
                    Set Lethal[l][a] = TRUE;
// Allele, a, of locus, l, is a recessive lethal.
                ELSE:
                    Set Lethal[l][a] = FALSE;
            END IF/ELSE
        END (founder allele) LOOP
    END (locus) LOOP
    Display initial population sizes on screen and
    write to output files;
END (population) LOOP
FOR (each year):
    IF (NumberPopulations > 1):
        GLOBAL_EV_RANDS();
// Generate random numbers for specifying environmen-
tal variation, concordant across populations, for the year.
        END (NumberPopulations) IF
        FOR (each population, p):
            LOCAL_EV_RANDS();
            CATASTROPHES(p);
// Determine if catastrophes occur that year.
        Determine carrying Capacity (K) for year;
// See Note 3.
        Add LocalKEVNRand * LocalKEV[p] to
        CarryingCapacity[p];
// Adjust K for local EV.
        Add GlobalKEVNRand * GlobalKEV[p] to
        CarryingCapacity[p];
// Adjust K for global EV.
        BREED(p);
// Go through breeding cycle to produce offspring.
        MORTALITY(p);
// Determine who dies that year.
        END (population) LOOP
        Add 1 to the age of each animal;
        IF (NumberPopulations > 1):
            MIGRATE();

```

```

// Determine which animals migrate between populations.
    END IF
    FOR (each population, p):
        IF (year during which animals are to be
            harvested):
            HARVEST(p);
        END (harvest year) IF
    END (population) LOOP
    FOR (each population, p):
        IF (year during which animals are to be
            supplemented):
            SUPPLEMENT(p);
        END (supplement year) IF
    END (population) LOOP
    FOR (each population, p):
        Tally PopulationSize[p];
        IF (population is not extinct AND
            population was not extinct prior year):
            // "Extinction" can be defined by the user as the absence of
            one sex, or as the population size falling below a specified
            lower limit.
             $r[p] = \log(PopulationSize[p] /$ 
             $PopulationSizePriorYear[p]);$ 
// Calculate population growth rate ( r ).
    END IF
    IF (not extinct AND PopulationSize[p], N
        > CarryingCapacity[p], K):
        FOR (each living animal):
            IF (RAND() > K / N):
// Stochastically kill excess above K.
                Animal dies;
            END IF
        END (each animal) LOOP
    END (N > K) IF
    Tally PopulationSize[p];
    IF (population is extinct):
        Decrement NumberExtantPopulations[p];
        IF (population was not extinct in prior
            year):
            Set YearExtinct[p] = CurrentYear;
            IF (population has not been
                recolonized):
                // First extinction
                    Set TimeToExtinction[p] =
                        CurrentYear;
                    Increment
                    NumberOfExtinctions[p];
                ELSE:
// Re-extinction of population
                    Set TimeToReextinction[p] =
                        CurrentYear -
                        YearOfRecolonization[p];
                    END (recolonized) IF/ELSE
                    END (was not extinct) IF
                ELSE:
// Not extinct
                    // Recolonization
                        IF (population was extinct in prior year):
                            Set YearRecolonized[p] = CurrentYear;
                            Set TimeToRecolonization[p] =
                                CurrentYear - YearExtinct[p];
                            Increment
                            NumberOfRecolonizations[p];
                        END (was extinct) IF
                        Set YearExtinct[p] = 0;
// Flag for not extinct
                        END (extinct) IF/ELSE
                        Display PopulationSize[p] on screen graph;
                    END (population) LOOP
                    FOR (each population, p):
                        CALC_GENETIC_METRICS(p);
                    END (population) LOOP
                END (year) LOOP
            END (iteration) LOOP
// At this point, the simulation is complete and summary
statistics can be calculated.
FOR (each population, p):
    Calculate and report means, SDs, and SEs across
    iterations for
        Population growth rate:  $r[p] = N[CurrentYear]$ 
        /  $N[PreviousYear]$ 
        TimeToExtinction[p]
        TimeToRecolonization[p]
        TimeToReextinction[p]
    FOR (each year):
        Calculate and output means, SDs, and SEs
        across iterations for:
            Probability of extinction, PE[p]
// SE = SQRT[PE * (1 - PE) / NumberIterations]
            PopulationSize[p]
            GeneDiversity[p]
// Gene Diversity = Heterozygosity expected under Hardy-
Weinberg equilibrium
            ObservedHeterozygosity[p]
// 1 - mean inbreeding coefficient
            NumberAlleles[p]
            LethalFrequency[p]
        END (year) LOOP
    END (population) LOOP
    Calculate and report within-population means of
    above summary statistics;
    Call program for displaying graphical displays of
    trends in:
        PopulationSize
        GeneDiversity
        Mean inbreeding coefficient (1 -
        ObservedHeterozygosity)
        Probability of population persistence to year
        Probability of extinction in that time interval
    Read in DoAnotherScenario?
    IF (DoAnotherScenario? is FALSE):
        BREAK from scenario LOOP

```

```

        END IF
    END (scenario) LOOP
END PROGRAM VORTEX()

BEGIN FUNCTION
READ_SPECIES_PARAMETERS():
// Get input parameters from keyboard or input file, describing simulation parameters, inbreeding effects, and basic species life history.
    Read in Input/Output file names;
    Read in NumberOfIterations;
    Read in NumberOfYears;
    Read in ExtinctionDefinition;
// Extinction can be defined as no animals of one sex, or as the population size falling below a specified minimum.
    Read in NumberOfPopulations;
    Read in InbreedingGeneticLoad;
    Read in ProportionLoadDueToLethals;
    Read in EVCorrelationBetweenReproductionAndSurvival?;
    IF (NumberOfPopulations > 1):
        Read in EVConcordanceAmongPoulations;
    END IF
    Read in NumberTypesOfCatastrophes;
    Read in Monogamous/Polygynous/Hermaphroditic?;
    Read in FemaleBreedingAge;
    Read in MaleBreedingAge;
    Read in MaximumAge;
    Read in SexRatio at birth;
    Read in MaximumLitterSize;
    Read in DensityDependentBreeding?
// The pseudocode for modeling density dependent breeding is not given below.
END FUNCTION READ_SPECIES_PARAMETERS()

BEGIN FUNCTION
READ_MIGRATION_PARAMETERS():
// Get input population structure and migration patterns.
    Read in MigrationAges;
    Read in MigrationSexes;
    Read in MigrationSurvival;
    Read in MigrationDensity;
    FOR (each population, pSource):
        FOR(each other population, pDestination):
            Read in MigrationProb[pSource][pDestination];
        END LOOP
    END LOOP
END FUNCTION
READ_MIGRATION_PARAMETERS()

BEGIN FUNCTION
READ_POPULATION_PARAMETERS(for population p):
    Read in ProportionFemalesBreeding[p];
    Read in BreedEV[p];
// Environmental variation is specified as a standard deviation.
        Read in Litter size distribution (either as MeanLitterSize[p] and SDLitterSize[p] or as the fully specified distribution of ProbLitterSize[p]/[n]);
        FOR (each age, x, up to FemaleBreedingAge):
            Read in FemaleMortality[p]/[x];
            Read in FemaleMortalityEV[p]/[x];
        END (age) LOOP
        FOR (each age, x, up to MaleBreedingAge):
            Read in MaleMortality[p]/[x];
            Read in MaleMortalityEV[p]/[x];
        END (age) LOOP
        FOR (each type of catastrophe, c):
            IF (NumberPopulations > 1):
                Read in GlobalOrLocal[p]/[c];
            END IF
            Read in CatastropheFrequency[p]/[c];
            Read in CatastropheBreedSeverity[p]/[c];
            Read in CatastropheSurvivalSeverity[p]/[c];
        END (catastrophe) LOOP
        CALC_DETERMINISTIC_GROWTH(p);
// Calculate deterministic population growth rate, generation time, and stable age distribution from mean birth and death rates. Effects of any catastrophes are averaged across years
        Read in ProportionMalesInBreedingPool[p];
// See Note 4.
        IF (initial numbers of animals are to be distributed according to the stable age distribution):
            Determine initial numbers of animals in each age-sex class;
// The stable age distribution would rarely assign whole numbers to each age-sex class. Integral numbers are assigned that most closely match the desired distribution.
        ELSE (does not start at stable age distribution):
            FOR (each sex):
                FOR (each age up to MaximumAge):
                    Read in initial number of animals;
                END LOOP
            END LOOP
        END (stable age distribution) IF/ELSE
        Read in CarryingCapacity[p] (K);
// K may be specified as a function of year or other parameters. See Note 3.
        Read in KEV[p];
        Read in Harvest[p]?
        IF (Harvest[p]? = Yes):
            Read in FirstYearHarvest[p], LastYearHarvest[p];
            HarvestInterval[p];
            FOR (each age, x, up to FemaleBreedingAge):
// For harvest, all adults are treated in the same age category.
                Read in NumberFemalesToBeHarvested[p]/[x];
            END LOOP
            FOR (each age, x, up to MaleBreedingAge):
                Read in NumberMalesToBeHarvested[p]/[x];
            END LOOP

```

```

END IF
Read in Supplement[p]?;
IF (Supplement[p]? = Yes):
    Read in FirstYearSupplementation[p],
    LastYearSupplementation[p],
    SupplementationInterval[p];
    FOR (each age, x, up to FemaleBreedingAge):
        Read in NumberFemalesToBeSupplemented[p][x];
    END LOOP
    FOR (each age, x, up to MaleBreedingAge):
        Read in NumberMalesToBeSupplemented[p][x];
    END LOOP
END IF
END FUNCTION
READ_POPULATION_PARAMETERS()

BEGIN FUNCTION
CALC_DETERMINISTIC_GROWTH(for population p):
// Use standard life table analysis; solve the Euler equation
to find the deterministic growth rate.
    Set fecundity, M = MeanLitterSize[p] * (1 - SexRatio);
// SexRatio is proportion males at birth.
    FOR (each type of catastrophe, c):
// Adjust M for catastrophes.
        Multiply M by CatastropheFrequency[p][c] *
        CatastropheBreedSeverity[p][c] + (1 -
        CatastropheFrequency[p][c]);
    END (catastrophe) LOOP
    FOR (each age, x):
        Set female survival, P/x = 1 - FemaleMortality[p][x];
        FOR (each type of catastrophe, c):
// Adjust P/x for catastrophes.
            Multiply P/x by CatastropheFrequency[p][c] *
            CatastropheSurvivalSeverity[p][c] + (1 -
            CatastropheFrequency[p][c]);
        END (catastrophe) LOOP
        Multiply cumulative survivorship, L/x, by P/x;
        IF (x >= FemaleBreedingAge):
            Add L/x * M to SumLxMx;
            Add x * L/x * M to SumAgeLxMx;
        END IF
    END (age) LOOP
    Set R0 = SumLxMx;
    Set GenerationTime[p] = SumAgeLxMx / SumLxMx;
// Preliminary estimate
    Set Lambda = R0 ^ (1/GenerationTime);
// Preliminary estimate
    Solve Euler equation by iterative approximation, to
yield precise Lambda;
    Set r[p] = log(Lambda);
    Set GenerationTime[p] = log(R0) / r[p];
    FOR (each age, x):
// Determine stable age distribution.
        Set StableAgeClassSize[p][x] = (1 - SexRatio) * L/x /
        (Lambda ^ x);
        Add StableAgeClassSize[p][x] to
        SumStableAgeClassSize[p];
    END LOOP
// Repeat age distribution calculations for males, but use
female-based Mx and Lambda.
    FOR (each age, x):
        Set male survival, P/x = 1 - MaleMortality[p][x];
        FOR (each type of catastrophe, c):
// Adjust P/x for catastrophes.
            Multiply P/x by CatastropheFrequency[p][c] *
            CatastropheSurvivalSeverity[p][c] + (1 -
            CatastropheFrequency[p][c]);
        END (catastrophe) LOOP
        Multiply cumulative survivorship, L/x, by P/x;
    END LOOP
    FOR (each age, x):
// Determine stable age distribution.
        Set StableAgeClassSize[p][x] = SexRatio * L/x /
        (Lambda ^ x);
        Add StableAgeClassSize[p][x] to
        SumStableAgeClassSize[p];
    END LOOP
    FOR (each age, x, and sex, s):
        Divide StableAgeClassSize[p][s][x] by
        SumStableAgeClassSize[p];
    END LOOP
END FUNCTION
CALC_DETERMINISTIC_GROWTH()

BEGIN FUNCTION GLOBAL_EV_RANDS():
    FOR (each type of catastrophe):
        Set GlobalCatastropheRand = RAND();
// Select random number to determine if global catastro-
phe occur. See Note 5.
        END (catastrophe) LOOP
        Set GlobalBreedEVRand = RAND();
// Select random number for specifying EV in breeding for
that year.
        Set GlobalBreedEVNRand = NRAND();
// Select random normal deviate for specifying EV in
breeding for year. Whether EVRand or EVNRand will be
used depends on the magnitude of EV. See Note 6.
        Set GlobalBreedEVNRand to same sign as
GlobalBreedEVRand;
        IF (EVCorrelationBetweenReproductionAndSurvival? =
No):
            Set GlobalMortEVRand = RAND();
// Select random 0-1 number for specifying EV mortality
for that year.
            Set GlobalMortEVNRand = NRAND();
// Select random normal deviate for specifying EV in mor-
tality for that year. Whether EVRand or EVNRand will be
used depends on the magnitude of EV.
            Set GlobalMortEVNRand to same sign as
GlobalMortEVRand;
            Set GlobalKEVNRand = NRAND();

```

```

// Select random normal deviate for specifying EV in K for
year.
ELSE (EV in breeding is correlated with EV in mortality):
  Set GlobalMortEVRand = GlobalBreedEVRand;
  Set GlobalMortEVNRand = GlobalBreedEVNRand;
  Set GlobalKEVNRand = GlobalBreedEVNRand;
END (EV correlation) IF/ELSE
END FUNCTION GLOBAL_EV_RANDS()

BEGIN FUNCTION LOCAL_EV_RANDS():
  Set LocalBreedEVRand = RAND();
// Select random number for specifying EV in breeding for
year.
  Set LocalBreedEVNRand = NRAND();
// Select a random normal deviate for specifying EV in
breeding.
  Set LocalBreedEVNRand to same sign as
  LocalBreedEVRand;
  IF (EVCorrelationBetweenReproductionAndSurvival? =
  FALSE):
    Set LocalMortEVRand = RAND();
// Select random number for specifying EV in mortality.
Set LocalMortEVNRand = NRAND();
// Select random normal deviate for specifying EV in mor-
tality.
  Set LocalMortEVNRand to same sign as
  LocalMortEVRand;
  Set LocalKEVNRand = NRAND();
// Select random normal deviate for specifying EV in K.
ELSE (EV in breeding is correlated with EV in mortality):
  Set LocalMortEVRand = LocalBreedEVRand;
  Set LocalMortEVNRand = LocalBreedEVNRand;
  Set LocalKEVNRand = LocalBreedEVNRand;
END (EV correlation) IF/ELSE
END FUNCTION LOCAL_EV_RANDS()

BEGIN FUNCTION CATASTROPHES(for population  $p$ ):
FOR (each type of catastrophe,  $c$ ):
  IF (Catastrophe is local in effect):
    IF (RAND() < CatastropheFrequency[ $p$ ][ $c$ ]):
// See Note 5.
      Set CatastropheFlag[ $c$ ] = TRUE;
// Catastrophe has occurred.
      ELSE:
        Set CatastropheFlag[ $c$ ] = FALSE;
      END (catastrophe) IF/ELSE
    ELSE:
      IF (GlobalCatastropheRand <
      CatastropheFrequency[ $p$ ][ $c$ ]):
        Set CatastropheFlag[ $c$ ] = TRUE;
      ELSE:
        Set CatastropheFlag[ $c$ ] = FALSE;
      END (catastrophe) IF/ELSE
    END (Local/Global catastrophe) IF/ELSE
  END (catastrophe) LOOP
END FUNCTION CATASTROPHES()

BEGIN FUNCTION BREED(for population  $p$ ):
// Find breeders for the year ...
FOR (each living animal in the population):
  IF (sex = female AND age  $\geq$  FemaleBreedingAge):
    Add female to breeding pool;
  END IF
  IF (not hermaphroditic):
    IF (sex = male AND age  $\geq$  MaleBreedingAge):
      IF (RAND() <
      ProportionMalesInBreedingPool[ $p$ ]):
        Add male to breeding pool;
      END IF
    END IF
  END IF
END (animal) LOOP
IF (no males selected for breeding pool, but adult males
do exist):
  Add one male at random to breeding pool;
END IF
IF (monogamous):
  FOR (each male in breeding pool,  $m$ ):
    Set MaleUsed[ $m$ ] = FALSE;
// Flag to indicate male is available for pairing
  END LOOP
END IF
IF (hermaphroditic):
  IF (only one breeding female AND
  ProportionSelfing[ $p$ ] = 0):
    EXIT BREED();
  END IF
END IF
FOR (each female,  $Dam$ , in breeding pool):
  Let BreedRand = RAND();
  GETBREEDRATE();
// BreedRate is probability of breeding for the female, given
by the user either as a constant, ProportionFemalesBreeding,
or as a function of population size and other parameters.
See Note 3.
  IF (BreedRate = 0):
    CONTINUE LOOP with next breeding female
  END IF
// Find a mate ...
  IF (hermaphroditic):
    IF (RAND() < ProportionSelfing[ $p$ ]):
      Let Sire = Dam
    ELSE
      Choose a Sire at random from breeding pool;
      WHILE (Sire is  $Dam$ ):
        Choose a new Sire;
      END WHILE
    END (selfing) IF/ELSE
  ELSE (not hermaphroditic):
    Choose a Sire at random from the male breeding
    pool;
END FUNCTION BREED()

```

```

IF (monogamous):
    WHILE (MaleUsed[Sire]):
        Choose a new Sire;
    END WHILE
    Set MaleUsed[Sire] = TRUE;
// Flag Sire as unavailable for future Dams
    END IF
END IF/ELSE
// Find the litter size for that pairing ...
IF (MaximumLitterSize > 0):
    Set CumulativeProbLitterSize[0] = 1 - BreedRate;
    FOR each possible litter size, n:
        Set CumulativeProbLitterSize[n] =
            CumulativeProbLitterSize[n - 1]
            + ProbLitterSize[p][n] * BreedRate;
    END LOOP
    FOR (each litter size, n, in decreasing order):
        IF (BreedRand > CumulativeProbLitterSize
            [n - 1]):
            Set LitterSize = n;
            BREAK from Litter Size LOOP
        END IF
    END LOOP
ELSE:
// MaximumLitterSize = 0 is a code for using normal distribution of litter sizes.
    Set LitterSize = MeanLitterSize[p] +
        SDLitterSize[p] * NRAND();
    Set LitterSize = max(0, LitterSize);
    Set LitterSize = min(2 * MeanLitterSize[p],
        LitterSize);
// Truncates symmetrically to avoid creating bias.
    Set IntegerLitter = Largest integer less than
        LitterSize;
    Set Remainder = LitterSize - IntegerLitter;
    IF (RAND() < Remainder):
// Round-off litter size probabilistically.
        Set LitterSize = IntegerLitter + 1;
    ELSE:
        Set LitterSize = IntegerLitter;
    END IF/ELSE
END IF/ELSE
// Create the offspring ...
    Set Inbreeding = Kinship between Sire and Dam;
    FOR (Offspring from 1 to LitterSize):
        Assign ID, age (= 0), population, alive (= TRUE);
        FOR (each of six loci):
// First locus is neutral, others can have lethals.
            Pick at random an allele from Dam;
            Pick at random an allele from Sire;
        END LOOP
        IF(not hermaphroditic AND RAND() <
            SexRatio):
            Assign sex as male;
        ELSE:
            Assign sex as female;
    END IF/ELSE
// Does offspring live? Offspring mortality is placed here in
// the code, rather than in the MORTALITY() function, for
// better speed and lower memory requirements.
    FOR (each non-neutral locus):
        IF (homozygous AND allele is a lethal):
            Offspring dies;
        END IF
    END LOOP
    IF (not yet dead):
        GETDEATHRATE();
        Set SurvivalRate = 1 - DeathRate;
        IF (Inbreeding > 0):
            Set SurvivalRate = exp(-0.50 *
                LethalEquivalents * Inbreeding);
        ENDIF
        IF (RAND() > SurvivalRate):
            Offspring dies;
        END IF
    END IF
    IF (not dead):
        Calculate kinship to every living animal;
// See Ballou (1983) for the method of calculating inbreeding
// and kinship coefficients.
        END IF
    END (offspring) LOOP
END (breeding females) LOOP
END FUNCTION BREED()

BEGIN FUNCTION GETBREEDRATE():
    Obtain BreedRate by evaluating fecundity function for
    population and individual parameters;
// Most often, the fecundity function will simply return
    ProportionFemalesBreeding entered by the user. VORTEX
    provides the option, however, of making breeding a function of
    PopulationSize, GeneDiversity, Inbreeding, and other
    variables. See Note 3.
    ADJUSTRATE(BreedRate, LocalBreedEV[p],
        LocalBreedEVRand, LocalBreedEVNRand);
// Adjust rate for local EV.
    ADJUSTRATE(BreedRate, GlobalBreedEV[p],
        GlobalBreedEVRand, GlobalBreedEVNRand);
// Adjust rate for global EV.
    FOR (each type of catastrophe, c):
        IF (CatastropheFlag[c] = TRUE):
            Multiply BreedRate by
                CatastropheBreedSeverity[p][c];
        END IF
    END LOOP
END FUNCTION GETBREEDRATE()

BEGIN FUNCTION MORTALITY(for population p):
    FOR (each living animal in the population):
        IF (age > 0):
// Infant mortality occurs within the BREED() function,
// not here.

```

```

IF (at maximum age):
    Animal dies;
ELSE:
    GETDEATHRATE();
    IF (RAND() < DeathRate):
        Animal dies;
    END IF
END IF/ELSE
END (age > 0) IF
END (animal) LOOP
END FUNCTION MORTALITY()

BEGIN FUNCTION GETDEATHRATE():
    Obtain DeathRate by evaluating mortality function for
    population and individual parameters;
// Most often, the mortality function will simply return the
    mortality rate entered by the user for the age and sex of the
    current animal. VORTEX provides the option, however,
    of making mortality a function of PopulationSize,
GeneDiversity, Inbreeding, and other variables.
    ADJUSTRATE(DeathRate, LocalMortEV[p],
LocalMortEVRand, LocalMortEVNRand);
// Adjust rate for local EV.
    ADJUSTRATE(DeathRate, GlobalMortEV[p],
GlobalMortEVRand, GlobalMortEVNRand);
// Adjust rate for global EV.
FOR (each type of catastrophe, c):
    IF (CatastropheFlag[c] = TRUE):
        Let DeathRate = 1 -
            (CatastropheSurvivalSeverity[p][c] *
            (1 - DeathRate));
    END IF
END LOOP
END FUNCTION GETDEATHRATE()

BEGIN FUNCTION ADJUSTRATE(Rate, EV, EVRand,
EVNRand);
    Determine binomial parameter n for modeling EV;
// See Note 6.
    IF (n < 26):
        // Find the adjusted Rate from binomial EV.
        FOR (each BinomialOutcome 0 through n):
            Add BinomialOutcome / n to BinomialProportion;
            Calculate BinomialProbability for
BinomialProportion;
            Add BinomialProbability to CumulativeBinomial;
            IF (EVRand < CumulativeBinomial):
                Set Rate = Binomial Proportion;
                BREAK from LOOP
            END IF
        END LOOP
    ELSE:
        // Use Normal distribution for EV, and truncate symmetrically
        // to avoid bias.
        IF (Rate > 0.5):
            Set UpperLimit = 1;
        Set LowerLimit = Rate - (1 - Rate);
    ELSE:
        Set UpperLimit = 2 * Rate;
        Set LowerLimit = 0;
    END IF/ELSE
    Add EV * EVNRand to Rate;
    Let Rate = max(LowerLimit, Rate);
    Let Rate = min(UpperLimit, Rate);
END IF/ELSE
END FUNCTION ADJUSTRATE();

BEGIN FUNCTION MIGRATE():
FOR (each living animal):
    IF (not in age range that migrates):
        CONTINUE LOOP with next animal;
    END IF
    IF (not a sex that migrates):
        CONTINUE LOOP with next animal;
    END IF
    Set MigrationRand = RAND();
    Set pSource to population of current animal;
    IF (MigrationRand > CumulativeMigrationProb
    [pSource][NumberPopulations] ):
        CONTINUE LOOP with next animal;
// Does not migrate
    END IF
    Obtain MigrationDensity by evaluating function, or
    using specified constant parameter;
// See Note 3.
    IF (PopulationSize[pSource] / CarryingCapacity
    [pSource] < MigrationDensity):
        CONTINUE LOOP with next animal;
    END IF
    Obtain MigrationSurvival by evaluating function, or
    using specified constant parameter;
// See Note 3.
// Find to which population the animal migrates ...
    FOR (up to 10 attempts to enter another
    population):
// The limit of 10 attempts is imposed to prevent an infinite
    loop from occurring when all populations are at carrying
    capacity.
        IF (RAND() > MigrationSurvival):
            Animal dies;
            BREAK from LOOP, CONTINUE with
            next animal;
        END IF
        FOR (each population, pDestination):
            IF (MigrationRand < CumulativeMigrationProb
            [pSource][pDestination] ):
                BREAK from LOOP;
// Animal will try to migrate to pDestination
        END IF
    END LOOP
    IF (Population pDestination at carrying capacity):
        IF (tried 9 times before to find an open population):

```

```

        Animal dies;
// Never found an open population into which to migrate.
        BREAK from LOOP, CONTINUE with
        next animal;
    END IF
    IF (CumulativeMigrationProb[pDestination]
    [NumberPopulations] = 0):
        Animal dies;
// Cannot migrate away from pDestination.
        BREAK from LOOP, CONTINUE with
        next animal;
    END IF
    Set MigrationRand = RAND();
    Set pSource = pDestination;
// Moves on from population pDestination , old pDestination
becomes current pSource.
    WHILE (MigrationRand >
        CumulativeMigrationProb[pSource]
        [NumberPopulations]):
        Set MigrationRand = RAND();
// Must migrate somewhere, so draw a new random
number.
    END WHILE
    END IF
END LOOP
Change animal's population to pDestination;
Adjust tallies of population sizes;
// Increment size of pDestination , decrement size of pSource.
END animal LOOP
END FUNCTION MIGRATE()

BEGIN FUNCTION HARVEST(for population p):
    FOR (each age, x):
// HARVEST() lumps all animal above breeding age as a
single class.
        IF (NumberMales[p][x] <=
NumberMalesToBeHarvested[p][x]):
            All males age x die;
        ELSE
            WHILE (number harvested <
NumberMalesToBeHarvested[p][x]):
                Choose at random a living male in age class x;
                Male dies;
            END WHILE
        END IF/ELSE
    END LOOP
    FOR (each age, x):
        IF (NumberFemales[p][x] <=
NumberFemalesToBeHarvested[p][x]):
            All females age x die;
        ELSE
            WHILE (number harvested <
NumberFemalesToBeHarvested[p][x] from age class):
                Choose at random a living female in age class x;
                Female dies;
            END WHILE
        END IF/ELSE
        Adjust tallies of population size;
    END LOOP
END FUNCTION HARVEST()

BEGIN FUNCTION SUPPLEMENT(for population p):
    FOR (each age, x, up to MaleBreedingAge):
        WHILE (number males created <
NumberMalesToBeSupplemented[p][x]):
            Create a male, assigning ID, age, sex, alleles,
population;
            Set kinships to all other animals = 0;
            Set Inbreeding = 0;
        END WHILE
    END LOOP
    FOR (each age, x, up to FemaleBreedingAge):
        WHILE (number females created <
NumberFemalesToBeSupplemented[p][x]):
            Create a female, assigning ID, age, sex, alleles,
population;
            Set kinships to all other animals = 0;
            Set Inbreeding = 0;
        END WHILE
    END LOOP
END FUNCTION SUPPLEMENT()

BEGIN FUNCTION CALC_GENETIC_METRICS(for
population p):
    FOR (each living animal in the population):
        Increment NumberAlleleCopies[a] for each of the
two alleles, a, at a neutral locus;
        IF (allele 1 is same as allele 2):
            Increment NumberHomozygotes;
        END IF
    END LOOP
    FOR (each allele, a, of the neutral locus):
        IF ( NumberAlleleCopies[a] > 0 ):
            Increment NumberExtantAlleles;
            Add 0.25 * (NumberAlleleCopies[a] /
PopulationSize[p]) * (NumberAlleleCopies[a] /
PopulationSize[p]) to ExpectedHomozygosity[p];
        END IF
    END LOOP
    Set GeneDiversity[p] = 1 - ExpectedHomozygosity[p];
    Set ObservedHeterozygosity[p] = 1 - (NumberHomozygotes /
PopulationSize[p]);
    FOR (each living animal in the population):
        FOR (each non-neutral locus):
            IF (allele 1 at the locus is a lethal):
                Increment NumberLethals;
            END IF
            IF (allele 2 at the locus is a lethal):
                Increment NumberLethals;
            END IF
        END locus LOOP
    END animal LOOP

```

```

Set LethalFrequency[p] = NumberLethals /
PopulationSize[p];
END FUNCTION CALC_GENETIC_METRICS()

```

Note 1: Random integers from 0 to 64K are generated by the algorithm given by Kirkpatrick and Stoll (1981). The C code was modified from Maier (1991). Random real numbers between 0 and 1 are produced by first generating a random integer between 0 and 64K, and then dividing that integer by 64K. Random numbers from a normal distribution, with mean = 0 and SD = 1 are generated by the polar algorithm supplied by Latour (1986). Binomially distributed numbers are generated by first calculating the cumulative probability distribution for the discrete outcomes of the desired distribution, then generating a random real number, and then assessing which binomial outcome covers the portion of the distribution encompassing the random real number.

Note 2: VORTEX asks for the effects of inbreeding to be entered as a number of “lethal equivalents” per diploid animal, with further specification of what proportion of this genetic load is due to recessive lethal alleles vs other genetic effects (such as overdominance). Recessive lethal alleles are modeled such that the death of animals homozygous for lethal alleles will reduce the frequency of the lethals and thereby reduce the average effects of future inbreeding. The proportion of inbreeding depression not due to lethal alleles is modeled as an impact on fitness that follows a negative exponential equation (Morton et al. 1956), and is not reduced during generations of inbreeding.

Note 3: For rates which can be specified as functions of age, sex, inbreeding, population size, gene diversity, year, and population, the rate to be used is determined by evaluating the function specified by the user. If the user enters a fixed constant for the rate, as is usually the case, then the function simply returns that constant. However, the user can specify a mathematical formula that defines a demographic rate as being density-dependent, or a function of other population parameters. For example, fecundity could be specified to decline in older age classes, adult mortality could be specified to increase with inbreeding, or habitat (carrying capacity) could be specified to decrease over time. The algorithms for parsing and evaluating user-defined rate functions (e.g., the first step of functions GETBREEDRATE() and GETDEATHRATE()) are not given in the pseudo-code.

Note 4: The proportion of males in the breeding pool can be entered directly, or indirectly in the form of the proportion of males that breed or as the average number of litters per breeding male. If the proportion in the breeding pool is given indirectly, VORTEX will assume that the distribution of male reproductive success follows a Poisson distribution. The proportion of males in the breeding pool is

then calculated by solving the following equations for the unknowns:

$$LittersPerMale = \text{ProportionFemalesProducingLitters} * \\ (\text{NumberAdultFemales} / \text{NumberAdultMales})$$

[Note: Adult sex ratio is determined from stable age distribution.]

$$LittersPerMale = \text{ProportionMalesInBreedingPool} * \text{LittersPerMaleInBreedingPool}$$

$$\text{ProportionMalesBreeding} = \text{ProportionMalesInBreedingPool} * (1 - \exp(-\text{LittersPerMaleInBreedingPool}))$$

This last equation adjusts for the fact that in any given year some males in the breeding pool will not happen to be successful breeders (the zero class of the Poisson distribution).

Note 5: The occurrence of probabilistic events is determined by a random number generator. The event is deemed to occur if a random number between 0 and 1 is less than the probability of occurrence for that event.

Note 6: Environmental variation (EV) in breeding and in each mortality rate is modeled as a binomial distribution or as a normal distribution, depending on whether the magnitude of EV is large. The user specifies a mean and standard deviation for each rate. The binomial distribution that has a standard deviation closest to the desired EV is determined by solving the equation for the binomial variance, $V = p * (1 - p) / n$, for the parameter n when given the mean, p , and variance, $V = SD^2$. The parameter n is then rounded to the nearest whole number. If $n < 26$, the binomial distribution with parameters p and n is used for EV. Because of the rounding step necessary to produce the discrete binomial distribution, this distribution will often have a slightly different variance than that entered by the user. If $n > 25$, the normal distribution with mean p and variance V will be used to model EV. In such cases, the normal distribution very closely approximates the binomial distribution.

The binomial distribution is restricted to the interval 0 to 1, and it fits well the distribution of demographic rates across years observed in some natural populations (e.g., Lacy 1993). The PVA program INMAT (Mills and Smouse 1994) uses the related beta distribution for this purpose, and it too is restricted to the biologically meaningful 0 to 1 interval. In contrast, the normal distribution extends infinitely in both directions, although the tails beyond 0 and 1 are typically very small in those cases for which VORTEX uses a normal distribution to model EV. For example, if $p = 0.5$ and $SD = 0.1$ (so that the binomial parameter $n = 25$, the limiting case for VORTEX to use the normal approximation), then the area of the normal distribution outside of the 0-1 range is < 0.000001 . When modelling EV as a normal distribution, the distribution must be truncated at 0 and 1. To avoid creating any bias in the mean demographic rate as a result of this truncation, VORTEX always truncates the distribution symmetrically.

For example, if the mean is $p = 0.3$, VORTEX truncates the distribution at 0.0 and 0.6. This truncation will cause the SD of the distribution to be very slightly less than that entered by the user.

Some PVA models use continuous distributions such as the normal or log normal to represent EV even when EV is large. In such cases, the necessary truncations can cause EV to be substantially less than intended by the user. Moreover, if the truncation is not symmetric, then the mean demographic rate generated by the model can be strongly biased away from the input parameter.

References

- Ballou, J. D. 1983. Calculating inbreeding coefficients from pedigrees. – In: Schonewald-Cox, C. M. et al. (eds), *Genetics and conservation: a reference for managing wild animal and plant populations*. Benjamin/Cummings, Menlo Park, California, pp. 509–520.
- Burgman, M. A., Ferson, S. and Akçakaya, H. R. 1993. Risk assessment in conservation biology. – Chapman and Hall.
- Caswell, H. 1989. Matrix population models: construction, analysis and interpretation. – Sinauer.
- Caughley, G. and Gunn, A. 1996. *Conservation biology in theory and practice*. – Blackwell.
- Kirkpatrick, S. and Stoll, E. 1981. A very fast shift-register sequence random number generator. – *J. Comp. Phys.* 40: 517.
- Lacy, R. C. 1993. VORTEX: A computer simulation model for population viability analysis. – *Wildl. Res.* 20: 45–65.
- Latour, A. 1986. Polar normal distribution. – *Byte*, August 1986: 131–132.
- Lindenmayer, D. B., Lacy, R. C. and Pope, M. L. 2000. Testing a simulation model for population viability analysis. – *Ecol. Appl.* 10: 580–597.
- Maier, W. L. 1991. A fast pseudo random number generator. – *Dr. Dobb's Journal*, May 1991: 152–157.
- Miller, P. S. and Lacy, R. C. 1999. VORTEX Ver. 8 users manual. A stochastic simulation of the simulation process. – IUCN/SSC Conservation Breeding Specialist Group, Apple Valley, Minnesota.
- Mills, L. S. and Smouse, P. E. 1994. Demographic consequences of inbreeding in remnant populations. – *Am. Nat.* 144: 412–431.
- Morton, N. E., Crow, J. F. and Muller, H. J. 1956. An estimate of the mutational damage in man from data on consanguineous marriages. – *Proc. Nat. Acad. Sci. USA* 42: 855–863.
- Pielou, E. C. 1977. *Mathematical ecology*. - Wiley.
- Starfield, A. M. and Bleloch, A. L. 1986. Building models for conservation and wildlife management. – MacMillan.

Considering threats to the viability of small populations using individual-based models

Robert C. Lacy

Lacy, R. C. 2000. Considering threats to the viability of small populations using individual-based models. – Ecol. Bull. 48: 39–51.

As wildlife populations become smaller, the number of interacting stochastic processes which can destabilize the populations increases: genetic effects (inbreeding and loss of adaptability) and instability of the breeding structure (sex ratio imbalances, unstable age distribution, and disrupted social systems) can decrease population growth and stability. Recent analyses have shown that some populations can be very sensitive to these stochastic processes, at larger population sizes than had been suggested previously, and often in unexpected ways. Interactions among processes can reduce population viability much more so than would be assumed from consideration of isolated factors. For example, in monogamous species, random fluctuations in sex ratio will depress the mean number of breeding pairs in populations with as many as 500 adults. At low population densities, individuals may not be able to find mates, or may not encounter individuals sufficiently unrelated to be accepted as suitable mates. Inbreeding depression of demographic rates can become a significant contributor to population decline in populations with several hundred individuals, even if genetic problems are not the primary threat. Most models of genetic decay in small and fragmented populations assume demographic stability. However, when the increases in demographic fluctuations of small populations are considered, rates of loss of genetic variation and accumulation of inbreeding can be much faster than has been suggested before. These processes can be examined in detailed, individual-based PVA models. Accurate data to parameterize these models, however, are often not available. Thus, we need to interpret cautiously PVA conclusions for populations that are small, highly fragmented, or projected for many generations.

R. C. Lacy (rldacy@ix.netcom.com), Dept of Conservation Biology, Daniel F. & Ada L. Rice Center, Chicago Zoological Society, Brookfield, IL, 60513 USA.

There are many kinds of threats to the viability of populations of wildlife. The processes which have driven many once-abundant populations down to one or few small populations in scattered remnants of habitat include direct exploitation (over-harvest), habitat destruction and fragmentation, degradation of habitat quality, introduction of exotic species, and chains of extinction (Caughley 1994). Often, after precipitous declines occur, conservation biologists and governmental agencies establish recov-

ery actions to try to prevent local extirpation of populations or the ultimate extinction of the taxon.

As wildlife populations become smaller, additional threats to stability and persistence arise, which can exacerbate the difficulty of stopping or reversing a decline. These problems of small populations generally result from stochastic or random processes. In any sampling process, the predictability of an outcome decreases as the sample size is reduced. Many aspects of population dynamics are inher-

ently sampling processes, rather than completely determined events, including: mate acquisition, breeding success, sex determination, transmission of genetic alleles, survival, and dispersal. The uncertainty in such processes can lead to instability in population dynamics. Moreover, fluctuations in demographic and genetic processes cause depression in long-term rates, because the geometric means and other appropriate compound measures of population performance are less than the arithmetic means. Finally, reductions in growth rates and fluctuations in rates can interact synergistically, causing increasing instability and more rapid decline, until the ultimate stability is reached when the population becomes extinct. These processes were termed “extinction vortices” by Gilpin and Soulé (1986), and their examination constitutes the core of most population viability analyses (PVA) (Soulé 1987, Boyce 1992, Lacy 1993/1994).

Caughley (1994) argued that there is a dichotomy in conservation biology between those who follow a “declining population paradigm,” examining deterministic causes of population decline, and those who follow a “small population paradigm,” examining the processes that further imperil populations after they have become small. Caughley called for more theory to guide the declining population approach, more data to support the small population approach, and better use of the strengths of the two approaches to guide conservation, but he also questioned whether too much emphasis has been given to small population processes in wildlife conservation. Hedrick et al. (1996) argued that the problems of small populations have at times been under-appreciated, but also that the processes causing population declines and the processes affecting populations that have become small are inter-linked in complex ways, so that PVA and conservation biology must encompass both of Caughley’s paradigms.

The problems of small populations have received extensive theoretical treatment (e.g., Soulé 1987), but further assessment of the factors affecting viability of very small populations is needed. I will argue that we often underestimate the importance of these factors in population viability, as the magnitude and even direction of some of these effects may be different than has been commonly supposed. Although I believe that the problems inherent in small populations are more numerous and more severe than is commonly recognized, the same may be true of the causes of population decline. However, because additional threats to population viability arise as populations become small, the kinds of PVA models that are needed for assessing the status and recovery options for a species will change as a species declines. Therefore, it is important to consider carefully which PVA model is most appropriate for a particular analysis (Lindenmayer et al. 1995, Akçakaya and Sjögren-Gulve 2000). An individual-based simulation program that models the stochastic processes of small populations in detail would probably not be the best model for examining viability of a population which numbers in the

tens of thousands. Similarly, a population-based structured model which ignores factors such as fluctuations in sex ratio, mate availability, and inbreeding would probably not be the most accurate model for a population which falls below 100 individuals. Many, perhaps all, presently used PVA models assess only some of the threats facing small populations, and therefore may underestimate probabilities of extinction and difficulties in species recovery.

In this paper, I will describe some of the threats to small populations that are not included in most PVA models. This discussion will provide guidance as to when more detailed, individual-based PVA models may be necessary to represent well the dynamics of small populations. Most of the processes I will discuss are particularly important for species with low intrinsic growth rates and stable social systems, and somewhat less so for those with high fecundity and little structure to the social or breeding system. Therefore, these considerations will be most applicable to mammal and bird species, and the bias toward these groups in examples below reflect my greater experience with PVA of these taxa.

Processes destabilizing small populations

Shaffer (1981) categorized the stochastic threats to small populations: demographic stochasticity, environmental stochasticity, natural catastrophes, and genetic stochasticity. These causes of uncertainty and fluctuation in population size interact, but they are conceptually distinct.

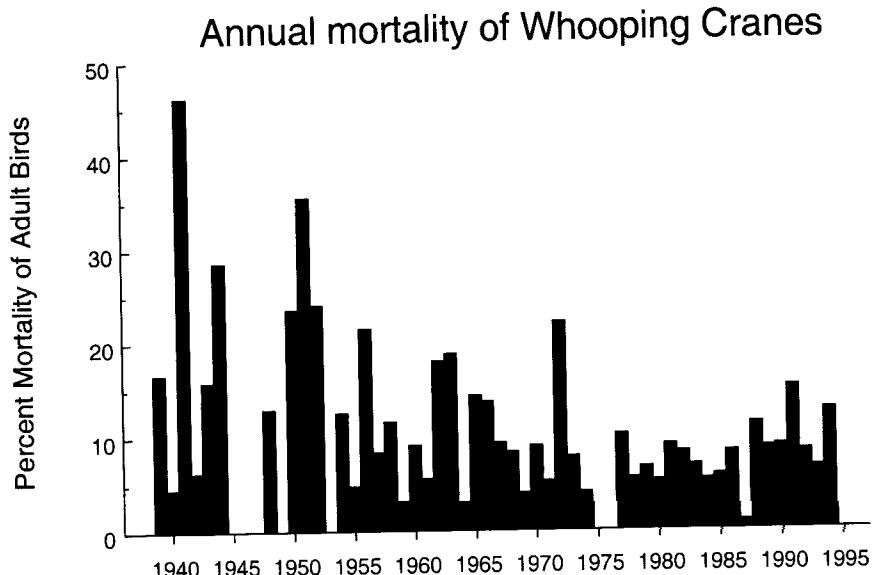
Demographic stochasticity

Demographic stochasticity is the random variation in the numbers of births, number of deaths, and sex ratio in a population that results from the fates of individuals being independent outcomes of probabilistic events of reproduction, mortality, and sex determination (Shaffer 1981). The observed variation across years or across populations with constant probabilities would be distributed as binomial distributions. If fates of individuals are independent, then demographic stochasticity is intrinsic to the population, and is a simple consequence of the sampling that occurs as individuals are subjected to the population rates.

Figure 1 shows the percent of whooping cranes *Grus americana* each year from 1938 to 1994 that failed to return the following year to the wintering grounds in Texas. Although some variation in mortality was due to environmental variation and likely catastrophes (see below), most of the variation in survival across years can be accounted for by demographic stochasticity (Mirande et al. 1991). The reduced variation in survival as the population grew in size from 18 birds in 1938 to ca 150 birds in the 1990s is clearly evident.

Demographic stochasticity has been recognized as a

Fig. 1. Percent of whooping cranes observed each year on the wintering grounds which did not survive to return the following year. In years with no bar shown, no birds died. Data from Mirande et al. (1991) and pers. comm.



potential threat to very small populations, but the contribution it can make to population instability has been underestimated. Based on simple calculations of the probability that all individuals will be of the same sex, or die synchronously, it has commonly been stated that demographic stochasticity can cause extinction only when populations fall below ca 10 – 20 individuals (Goodman 1987, Shaffer 1987). However, processes that are not serious problems when acting alone can become significant contributors to population instability and decline when they act synergistically with other threatening processes. The last five dusky seaside sparrows *Ammospiza maritima nigrescens* were all males, an unfortunate but not overly surprising event that effectively eliminated the taxon. The probability that a population of 100 breeders would all be males is essentially zero, so it might be assumed that random fluctuations in sex ratio are unimportant in populations approaching such a size. However, fluctuations in sex ratio can depress population growth significantly, even in such cases. In a population of 100 breeders, we expect ca 50 females and 50 males. But due just to demographic stochasticity, the number of females will deviate by 5 or more (one standard deviation) from this expectation in about one-third of the years. In monogamous species, such as most birds, this means that typically < 50 pairs could be formed. The mean depression in reproduction relative to a population with a constant equal sex ratio can be calculated from the mean absolute deviation of the binomial distribution. Due solely to the random fluctuations in sex ratio, reproduction in a monogamous population with 100 adults would be depressed by ca 8%. This level of reduced population productivity is enough to cause low-fecundity species to switch from positive population growth to long-term population decline and eventual extinction. Brook et

al. (1999) found that interaction of the breeding system with fluctuations in the sex ratio strongly influenced projections for population growth of whooping cranes.

Figure 2 shows the mean depression in breeding caused by random variation in the sex ratio for monogamous populations of various sizes. Even with 500 breeding individuals, the mean number of pairs in the population is 3.6% below what would be available if the sex ratio were fixed at 50:50. This simple example of a threat to the viability of small populations illustrates several important points. First, the random deviation in sex ratio does not in itself cause extinction except in the very smallest of populations, but it can interact with other factors such as the breeding system to depress population growth in a vulnerable population sufficiently to cause extinction. Second, it is unlikely that biologists observing the population would recognize that fluctuation in the sex ratio was a contributing cause of lower reproduction and population decline. Third, it would be possible to incorporate the reduction in breeding as an average effect in a simple life table projection. However, to do so requires that the demographic rates were estimated from a population of the size of the population being currently assessed, and that the population remains constant in size. (This last assumption defeats the purpose of PVA.) The effect of biased sex ratios depressing reproduction in monogamous population of changing size could be modeled as a density-dependent effect on reproduction (Stephens and Sutherland 1999, Courchamp et al. 1999). However, I am not aware of any cases in which an analytical or population-based PVA model incorporated the reduction in breeding, as shown in Fig. 2, for monogamous species that would be expected due to sex ratio fluctuations. Many population models ignore sex ratio and breeding system entirely, projecting numbers of females

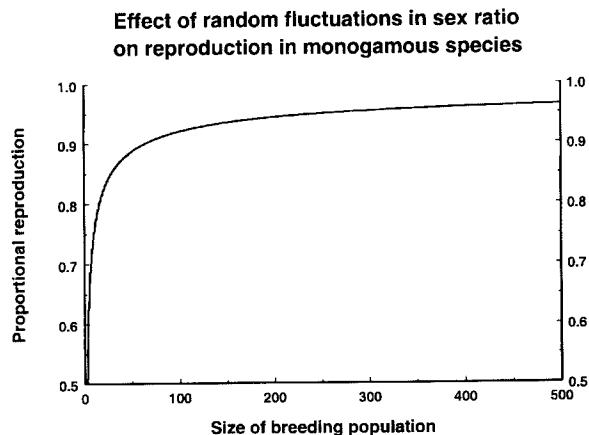


Fig. 2. Mean proportion of monogamous pairs that could be formed relative to the case of a constant 50:50 sex ratio, as a consequence of random fluctuations in the sex ratio in breeding populations of varying size.

under the assumption that there are always males available for mating. Individual-based models are well suited for cases in which sex-ratio biases can disrupt breeding, because they automatically generate stochastic variation in the sex ratio. Rules defining the breeding system can then be built into the model.

Thus, if there is not promiscuous breeding, random fluctuations in the sex ratio can depress population growth in even moderate-sized populations. Similarly, random demographic stochasticity in the numbers of births and deaths per year can depress mean population growth, because of variation in the age distribution and other disruptions of optimal breeding. For example, managers of zoo populations are often distressed to find that reproduction is kept well below optimal levels because of temporary imbalances in the sex ratio or the age distribution. Following the advice of conservation biologists, zoo managers have assumed that a population of 50 or more is safe from the threat of demographic stochasticity, but random fluctuations are causing problems for maintaining stable populations of rhinoceroses, spectacled bears, lions, and other species for which there is limited flexibility to accommodate changes in numbers.

Environmental variation

Environmental variation or stochasticity is the variation in demographic rates or probabilities that results from fluctuations in the environment (Shaffer 1981). Thus, local environmental variation causes temporal clustering of births and deaths, which would increase uncertainty and variability in population size, and thereby make a small population more vulnerable to extinction. The kinds of perturbations of the environment which cause variation in birth and death rates include disease, sporadic predation,

irregular food availability, and variable weather. Natural catastrophes are the extreme of environmental variation, in which droughts, floods, fires, disease epidemics, and other local disasters can decimate a population.

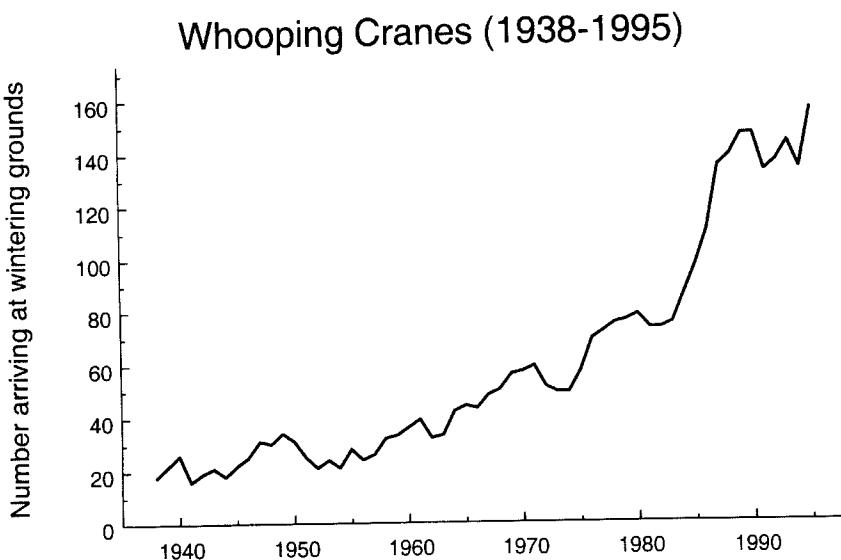
Although both demographic stochasticity and environmental variation cause fluctuations in the number of births and deaths in a population, the processes are conceptually distinct and statistically independent. Demographic stochasticity is intrinsic to all populations, regardless of the stability of the environment. As a binomial sampling process, it is highly dependent on the population size. Environmental variation results from variation in habitat quality over time, and is unrelated to population density. The variance in demographic rates caused by environmental variation would be additive with variation due to demographic stochasticity (Goodman 1987).

Environmental variation is not usually affected by the local size of the wildlife population, except in those cases, such as predator-prey interactions, in which the organisms have large effects on their local environment. However, the threat to population viability caused by a given level of environmental variation would be more severe in smaller populations, because smaller populations are closer to extinction. Moreover, the amount of environmental variation would be highly dependent on the total area of habitat occupied by a population. Many environmental stresses are localized, so a population exploiting a large area would benefit from the averaging of any environmental fluctuations that are not synchronous over the entire range. Individuals might use spatial variation in environmental conditions to allow escape from temporal variation in the environment (Kindvall 1996). Even if individuals do not move away from areas with temporarily poor conditions, temporary population declines in some areas would be offset by growth elsewhere.

In PVA models, consideration should be given to whether the amount of environmental variation in the system should change with range contraction and expansion. Unfortunately, data on variation in demographic rates are woefully inadequate for almost all species. Usually, we have no more than crude guesses as to the magnitude of environmental variation for use in population viability analyses. There would be considerable value in a compilation of data across species which would allow generalizations concerning the typical magnitude of fluctuations in demographic rates for species with various life histories, trophic guilds, and habitat types.

Figures 3 and 4 show two examples of fluctuations in natural populations that contrast markedly in the extent to which they are impacted by environmental variation. The population trend for whooping cranes during recovery from a near-extinction shows the reduced relative fluctuations in numbers as the population increased in size. Cranes form long-term monogamous pair bonds, they return to nesting sites for a number of years, they have low fecundity, and they are very long-lived. Hence, they would

Fig. 3. Number of whooping cranes arriving each winter at the Aransas National Wildlife Refuge, Texas. Data from Mirande et al. (1991).

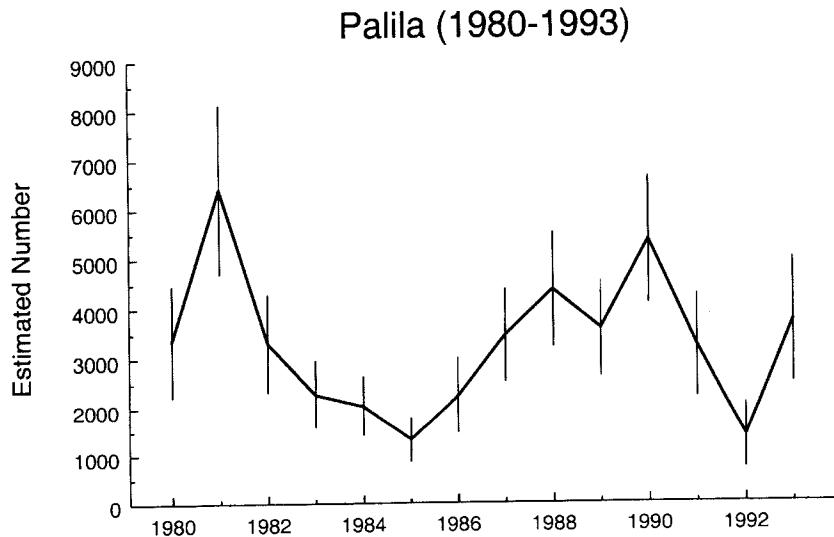


be expected to show minimal susceptibility to environmental variation. Observed variation in the numbers of hatchlings from 1976 to 1991 was approximately that which would be expected due solely to the demographic variation that would result from a random annual sampling of breeders from the pool of adult birds, each with a constant probability of reproductive success (Mirande et al. 1991). Variation across years in first-year survival was more than three-fold greater than what would be expected due to demographic sampling, indicating that the probability of chick mortality fluctuates across years due to environmental variation. Adult mortality, however, seemed to fall into two classes: in some years, mortality was significantly greater than mortality in other years and deviated significantly further from the mean than can be expected from random demographic stochasticity. The cause of

these years of poor survival is not known, as census counts were made only once a year during those years. However, they would appear to be examples of natural catastrophes. If these catastrophe years are excluded from the data as special cases, the annual variation in adult mortality was only slightly greater than that expected due to random demographic stochasticity (Mirande et al. 1991).

Figure 4 shows the population trends for the palila *Loxoides bailleui*, a finch which is restricted to the manane forests on the slopes of the major volcanoes of the island of Hawai'i. Although some variation in population size may be caused by imprecise census estimates, the species clearly undergoes striking fluctuations in numbers, even though the mean population size is ca 30 times greater than the current population of whooping cranes. Palila must be sensitive to environmental variation, probably

Fig. 4. Estimated numbers of palila, with 90% confidence intervals, from field surveys. Adapted from Ellis et al. (1992).



with respect to both breeding and survival. In part because of the much higher sensitivity to environmental variation, PVA modeling projected a higher probability of extinction for palila (Ellis et al. 1992) than for the much smaller population of whooping cranes (Mirande et al. 1991). The contribution of environmental variation to extinction vortices is well-recognized (Belovsky 1987, Goodman 1987, Foley 1994), but as with demographic stochasticity, we may not always recognize how large the effect can be.

Disrupted breeding systems

There is another cause of demographic instability in small populations, but it does not fit easily within the categories of Shaffer (1981), is rarely considered in PVA models, and may not be fully recognized in conservation plans. At low population densities, the social systems of many species may be disrupted. Such Allee effects are one type of density dependence in reproductive success. Although the importance of understanding and modeling density dependence has been stressed by some authors (Mills et al. 1996, Brook et al. 1997), most of the attention has been given to changes in demographic rates as the population approaches carrying capacity. Yet it is not when a population is near carrying capacity that we need to be concerned about extinction. Allee effects at the low end of density can be incorporated into several of the widely available "generic" PVA models (e.g., RAMAS/Space: Akçakaya and Ferson 1992; VORTEX: Lacy 2000), but this feature seems rarely to be used. Individual-based models can be tailored to provide detailed representations of specific social systems, and this approach was used to look at how the interactions of stochastic processes and pack structure impact viability of wolves (Vucetich et al. 1997) and wild dogs (Vucetich and Creel 1999).

Disruptions of the breeding system can occur for reasons that range from the obvious to the subtle. At very low population densities, animals may be unlikely to encounter any potential mates when they are ready to breed, and non-selfing plants may not be adequately pollinated (Menges 2000). Sumatran rhinoceroses *Dicerorhinus sumatrensis* now exist at densities of only a few per protected area. Field surveys in Malaysia in 1995 found tracks of one juvenile among 35 sets of tracks, and one of 21 adult females captured in the prior decade was pregnant (AsRSG. 1996). If the population were breeding as expected for a rhinoceros species, ca 30% of adult females should be pregnant at any time, and ca 15% of the animals should be under two years of age. Although detailed studies of demography have not been carried out (in part because of the difficulty of studying secretive animals that are at low density in the forest), it is plausible that the scarcity of mates is causing a near cessation of breeding over much of the fragmented range.

Even if some potential mates are available, small popu-

lations may provide little opportunity for mate choice. The extent to which a reduced pool of possible mates may be causing breeding delays or failures in small natural populations has not been explored. This problem could be exacerbated if the potential mates are all closely related to the choosing individual or to each other. Ryan (2000) found that, when given a choice of unfamiliar, distantly related females in a Y-apparatus, male *Peromyscus polionotus rhoadisi* mice preferred the less-related female even when differences in kinship averaged only $f = 0.013$, about the level of second cousins. When subsequently paired with one or the other female from the choice test, breeding was delayed and litter sizes were smaller when the only available partner was the one that had been less preferred.

Species with complex social systems may be especially vulnerable to problems resulting from low population density. For example, striped-back wrens *Campylorhynchus nuchalis* have very low breeding success unless they have at least two adult non-breeding helpers sharing in defense against nest-predators, and breeding success is related strongly to the number of such helpers (Rabenold 1990). If such a population were to decline in numbers, recruitment could stop when few birds remained to serve as helpers. One population was rescued from demographic decline when immigrants from nearby populations joined remnant breeding groups (Rabenold et al. 1991). Presumably, extirpation of the local population would have occurred if there had not been nearby sources of immigrants.

Inbreeding depression and loss of genetic diversity

At least two kinds of genetic problems can impact the viability of small populations: reduction of fitness of individuals resulting from inbreeding, and loss of genetic diversity due to random genetic drift (Lacy 1997). There has been much written about (e.g., Frankel and Soulé 1981, Schonewald-Cox et al. 1983, Hedrick and Miller 1992, Frankham 1995a), but also much debate over (e.g., Lande 1988, 1995, Caughley 1994, Caro and Laurenson 1994, Hedrick et al. 1996), the importance of genetics to conservation of wildlife populations.

Inbreeding depression, the reduction in fitness of inbred individuals, frequently occurs when normally out-crossing organisms mate with close relatives. It is commonly believed that it is a problem only for captive populations (which can be substantially buffered from many other risks facing small populations) and for a very few and very small natural populations, or perhaps only as a transient problem that would diminish as selection removes deleterious alleles during repeated generations of inbreeding. Yet, increasing numbers of studies are showing that inbreeding depression can impact population viability to a greater extent, more quickly, and less reversibly than previously supposed (Frankham 1995b, Lacy 1997). Jiménez et al.

(1994) found that inbreeding caused much lower survival of white-footed mice *Peromyscus leucopus* that had been released into a natural habitat than would have been predicted from laboratory measures of inbreeding depression. A population of the greater prairie chicken *Tympanuchus cupido pinnatus*, which had suffered a demographic decline from ca 25 000 birds to < 50 birds over 60 yr, consequently lost substantial genetic variability and suffered reduced fertility and egg viability from inbreeding depression (Westemeier et al. 1998). In Sweden, inbreeding depression of fertility may have been the cause of the rapid decline to extinction of a population of 15 – 20 pairs of the middle spotted woodpecker *Dendrocopos medius* which had been isolated 30 yr before (Pettersson 1985). In Finland, Saccheri et al. (1998) found that local populations of the Glanville fritillary butterfly *Melitaea cinxia* with lower heterozygosity, indicative of greater inbreeding, had lower egg-hatching rate, larval survival, and adult longevity. Apparently as a consequence, these populations had much higher probabilities of extinction.

Following Franklin (1980) and Soulé (1980), only populations with effective sizes below ca 50 have been commonly perceived to be at risk of significant inbreeding depression. To have an effective population size of 50, a typical natural population of a large mammal might need a total population size of ca 200. It is worth reconsidering the likely cumulative effects of inbreeding on the viability of such a population. Inbreeding would accumulate at a rate of 1% per generation. After 10 generations, the 10% cumulative inbreeding may cause a 5% – 20% reduction in survival and in fecundity (Ralls et al. 1988, Lacy et al. 1996, Lynch and Walsh 1998). The consequent reduction in population growth would be sufficient to cause low-fecundity species to decline. Yet, many wildlife managers with responsibility for populations of approximately this size assume that they can ignore effects of inbreeding, and most PVA models for populations of such size omit the impacts of inbreeding on demography.

Many conservation biologists assume that slow inbreeding will not reduce fitness, because selection can remove deleterious alleles during generations of inbreeding (Charlesworth and Charlesworth 1987). Yet, experimental evidence shows that such purging of the genetic load of deleterious alleles from a population often does not work: many populations continue to decline in fitness as they become increasingly inbred (Ballou 1997, Lacy and Ballou 1998) and may go extinct as a consequence (Frankham 1995b). Theoretical work indicates why selection is often ineffective in reducing inbreeding depression. At the small population sizes at which inbreeding occurs, random genetic drift is a much larger force in determining which alleles increase or decrease in frequency than is all but the strongest selection: random loss of adaptive alleles is almost as likely as loss of the deleterious alleles. Except when inbreeding depression is due primarily to a few highly deleterious recessive alleles, inbreeding is more likely to lead

to population extinction than to significant reduction of the genetic load (Hedrick 1994). Some populations may be fortunate enough not to carry a genetic load of deleterious alleles which would be expressed under inbreeding, but the evidence suggests that sensitivity to inbreeding may be determined by chance events such as founder effects as much as by any predictable factors (Ralls et al. 1988, Barrett and Kohn 1991, Lacy et al. 1996).

The rate of adaptive evolution of any population is expected to be proportional to its additive genetic variation (and heritability) for the traits under selection (Fisher 1958), and the limited evolutionary potential of domesticated animals with depleted genetic variation has been shown repeatedly. It would be difficult to know whether limited response to selection in rapidly changing habitats is a contributing threat to the persistence of natural populations. Some populations have persisted at small numbers for many generations (e.g., the Javan rhinoceros *Rhinoceros sondaicus*, and, until 1986, the black-footed ferret *Mustela nigripes*), but other populations have rapidly gone extinct when an unusual stress appeared in the environment (e.g., the golden toad *Bufo periglenes* [Pounds and Crump 1994], and the black-footed ferret in 1986 [Clark 1989]). Given the current situation of unprecedented environmental change and an accelerating extinction rate, perhaps more PVA models should consider the maintenance of sufficient genetic variability to ensure ecological and evolutionary flexibility, rather than solely immediate fitness effects and short-term population persistence. One way to accommodate long-term viability into conservation planning would be to use the potential for rapid recovery as the primary measure of population viability, rather than mere population persistence at small numbers. It is also possible to use models that include both the effects of inbreeding on demographic rates and the effects of reduced genetic variability on vulnerability to environmental variation and ability to survive catastrophes. The individual-based model VORTEX (Lacy 2000) can accommodate such dependencies on inbreeding, and similar effects could be built into population-based matrix models as in INMAT by Mills and Smouse (1994). Parameterization of models with genetic-demographic interactions is difficult, but data are increasingly available on the effects of inbreeding on demographic rates (Ralls et al. 1988, Brewer et al. 1990, Saccheri et al. 1998), persistence through environmental stress (Miller 1994, Keller et al. 1994), and population extinction (Frankham 1995b, Saccheri et al. 1998).

Interactions among threatening processes

Although each process described above can individually threaten the viability of small populations, synergistic interactions exacerbate the impacts of many of the processes and are at the center of extinction vortices (Gilpin and Soulé 1986). Interactions among threats are sufficiently

complex that they are often omitted from analytical PVA models and from the functions driving demographic projections in simulation models. Most analytical models are constructed by deriving the impact that a factor would have when acting in isolation from other threatening processes. Below are a few examples of interactions among threatening processes that can reduce population viability, often in unexpected and unexpectedly strong ways.

Increased dispersal among patches of habitat is usually assumed to help stabilize a metapopulation. Increased dispersal can restore genetic variation to previously inbred populations, can reduce demographic fluctuations within local populations, can rescue demographically weak populations (Brown and Kodric-Brown 1977), and can lead to recolonization of temporarily extirpated local populations. However, the metapopulation dynamics of small, partly isolated, and frequently extirpated populations can be highly dependent on spatial, temporal, and behavioral aspects of the population structure (Fahrig and Merriam 1994). For example, if a metapopulation declines to a level at which many of its constituent local populations are very small or extinct, then the benefits of dispersal can be replaced by disadvantages. Uncompensated emigration from isolated populations can depress local population growth (Fahrig and Merriam 1985), and suitable habitat that is temporarily empty can act as a population sink where animals fail to find mates (Gyllenberg and Hanski 1992). Consequently, increased dispersal can accelerate decline of sparsely populated metapopulations (Lindenmayer and Lacy 1995).

This collapse into a metapopulation vortex may be more likely if dispersal behaviors evolved in a physical and biotic environment that was different from the current landscape. For example, in largely contiguous habitat with local competition for breeding territories, the optimal dispersal pattern might be for subadults to always disperse, and to disperse in a random direction. When habitat is highly fragmented and often unoccupied, however, it might be adaptive to remain near the natal site unless local densities are very high (Ronce et al. 2000), and to develop dispersal behaviors that more efficiently locate suitable habitat (for example, habitat with an excess of inhabitants of the opposite sex). Many metapopulations may be occupying recently fragmented landscapes for which their evolved dispersal strategies are suboptimal. In PVA models, it might be important to consider that dispersal strategies that are stabilizing at one population density can become destabilizing at different population densities.

Disruption of breeding systems represents another example of interactions between processes. As populations become small, individuals may become closely related to most or all potential mates. In a number of species, individuals have been observed to avoid mating with genetic relatives (Keane 1990). Inbreeding avoidance in a small population could lead to frequent failures to locate any suitable mates. The suppression of breeding might be con-

sidered a form of inbreeding depression, but it may not be recognized as such, since the individuals may all have genotypes that otherwise would confer high fitness, and individuals may not be inbred (in fact, it is the avoidance of inbreeding that causes the depression of fitness). Genetic homogeneity leads to an epiphenomenon, with frequency dependent selection causing the "inbreeding depression" at the population level. Some long-isolated populations of beach mice (such as *Peromyscus polionotus leucocephalus*) have low genetic diversity, low frequencies of breeding when mice are paired in captive colonies, and the poor breeding is increasingly exacerbated when experimental populations are further inbred (Brewer et al. 1990, Lacy and Ballou 1998). It is possible that mate choice behavior that evolved to prevent inbreeding is now often preventing breeding, as the mice breed readily when paired with mice from other subspecies (Lacy unpubl.).

Inbreeding interacts with other threats to population viability. For example, Keller et al. (1994) found that inbred song sparrows *Melospiza melodia* were less likely to survive a severe winter, and inbred animals have occasionally been observed to be more vulnerable to other environmental stresses (Miller 1994).

Not only does demographic decline increase inbreeding, which can in turn further depress mean demographic rates, but smaller populations undergo greater relative demographic fluctuations. The increased fluctuations in numbers depress the genetically effective population size (N_e), as inbreeding and genetic drift are more rapid in a population that fluctuates in size than in a stable population of the same mean size. Together with other factors which can reduce breeding success in smaller populations, this can cause the ratio of the effective population size to the total size (N_e/N) to diminish as a population becomes smaller. Hence, while N_e might be 500 when $N = 1000$, N_e may be 30 when $N = 100$, and just 10 when $N = 50$.

In yet another example of the interaction between genetic and demographic threats to population viability, the joint effects of stochasticity in these two processes have been found to lead to consequences which can be opposite those predicted from purely genetic models. A number of authors have reported that a subdivided population will retain more gene diversity over time than will a single panmictic population, because genetic drift will by chance favor different alleles in the various isolated populations (Boecklen 1986, Varvio et al. 1986, Lacy 1987). However, the models on which this conclusion is based assume that populations are constant in size. The apparently beneficial effect of subdivision is partly due to an artifact of the models, in which model constraints create more equal distribution of reproductive success (and therefore higher N_e) when the population is divided into subunits of fixed size (Barton and Whitlock 1997), and partly due to the protection of different alleles as each subpopulation becomes fixed for a random subset of the diversity of the original metapopulation. This second process can be a benefit of

population subdivision, but it becomes significant only after populations lose most of their gene diversity and become highly inbred, and it depends on no populations becoming extinct as a result of that inbreeding or other factors.

In individual-based PVA models that model genetic changes, the effects of subdivision on gene diversity are quite different, because the PVA models do not unrealistically constrain the populations to be constant in size. Smaller populations undergo greater fluctuations in number and therefore lose gene diversity much faster than if they were part of a larger breeding population. As a consequence of this greater demographic instability, fragmented metapopulations will usually lose genetic variation (both heterozygosity and number of alleles per locus) more rapidly than does a single, more panmictic, large population. This trend occurs even when the effects of fragmentation are partly offset by dispersal among partly isolated populations. In models of populations of mountain brush-tail possums *Trichosurus caninus*, Lacy and Lindenmayer (1995) found that both heterozygosity and number of alleles were reduced more quickly when metapopulations of 100 or 200 possums were fragmented into 2, 5, or 10 subpopulations that exchanged up to 5% of their residents each year. Rates of genetic decay slowed as dispersal rates were increased, but the individual-based stochastic model consistently showed that population subdivision caused faster genetic decay from the metapopulation – a result not predicted from many analytical genetic models.

Characteristics of highly vulnerable species

Summarizing this discussion of the threats to viability of small populations, we can identify some of the characteristics of populations that lead to the most complicated population dynamics as populations become small, and which therefore might require the most detailed and individual-based PVA models. Especially vulnerable species would include those with non-breeding helpers, such as striped-back wrens and naked mole rats, species with cooperative foraging, such as many parrots and social carnivores, and species with group defense behaviors, such as musk ox and many primates. Species with precise mechanisms for mate choice, such as many bird species, could have demographic and genetic problems when that choice becomes limited. Monogamous species will have depressed reproduction when there is demographic stochasticity in the sex ratio.

Species with low fecundity are particularly vulnerable to inbreeding depression (Mills and Smouse 1994), because they can withstand less depression of survival before population growth rates become negative and because they will recover more slowly from population bottle-

necks. Moreover, of the few PVA models which consider genetic effects (e.g., INMAT: Mills and Smouse 1994; VORTEX: Lacy 2000; and see Menges 2000 for references on plant PVAs), usually an assumption is made that individuals at the start of the population projection are all unrelated and noninbred. Thus, we may under-appreciate existing or imminent genetic problems in populations which have already lost much genetic variation. For example, early analyses of the remnant population of the Florida panther *Felis concolor coryi* assumed that prior inbreeding would not diminish reproductive rates (Seal and Lacy 1989), even though a majority of the males had only one or no functional testicles (Seal et al. 1992). Golden lion tamarins *Leontopithecus rosalia rosalia* are now restricted to very small remnants of the original Atlantic coastal forest of Brazil, have been reduced to a population of ca 350 animals and scattered smaller populations (Ballou et al. 1998), have low genetic diversity (Forman et al. 1986), and show significant depression of juvenile survival when inbred (Dietz et al. 2000). Thus, any PVA models that assume tamarins presently have adequate genetic diversity may project, perhaps incorrectly, that the population could lose much more variation before genetic problems began to reduce population viability.

PVA modeling of small population processes

As illustrated in the examples presented above, more processes threaten the viability of small populations than are commonly addressed in PVA models. Analytical models and simple population projection models can therefore overestimate population growth, underestimate population fluctuations, and seriously underestimate probabilities of extinction. More accurate PVA of small populations may often require individual-based models that simulate interactions among threatening processes rather than relying on theoretical equations derived under assumptions of simplified population processes acting in isolation. Lindenmayer et al. (2000) found that PVA model predictions matched the observed dynamics of populations of three species of marsupials in a highly fragmented landscape only when the models incorporated distance-dependent and density-dependent dispersal, high mortality during dispersal, and spatial variation in habitat quality. Knowledge of times since isolation of each habitat fragment was also critical, as the metapopulations were projected to lose more component subpopulations before reaching extinction-recolonization equilibria. In an alternative approach, Sjögren-Gulve (1994, Sjögren-Gulve and Ray 1996) used a logistic regression model to incorporate similarly detailed information about the habitat characteristics, spatial arrangement, and surrounding forestry practices on the extinction-recolonization dynamics of pool

frog *Rana lessonae* metapopulations. Rather than modeling causal processes, as is done in individual-based models, logistic models can use observed correlates of population transitions to generate predictive models of metapopulation trends. Individual-based models require detailed data on the factors driving population processes, while logistic transition-incidence models require detailed data on the important correlates of population transitions over significant periods of time.

Generalized analytical models can be extremely valuable for discerning many broad trends (e.g., Belovsky 1987), but they do not provide the situation-specific representations that are needed to assess local threats to specific small natural populations nor, usually, the time-specific projections that are needed to understand non-equilibrium systems. Thus, detailed and often individual-based models are more appropriate for comparing management options for endangered species recovery and local conservation planning. We should also retain some skepticism regarding the generality of theoretical results until they have been confirmed to apply to simulated (or, better, real) populations in which the unrealistic assumptions of the theoretical model have been relaxed. There have been recent encouraging confirmations of the ability of PVA models to predict population dynamics (Brook et al. 2000), but more comparisons are needed among analytical results, simulation results from models of varying detail, experimental data, and observations on natural populations.

For a variety of reasons, PVA models for small populations may need to be highly specific with respect to how they model breeding systems, dispersal behavior, and genetic processes. Simple generalizations of population genetics theory may be misleading, because most of that theory was based on large sample approximations. (For example, generalizations about effects of dispersal among small populations often have assumed that an infinite number of

such populations exist.) Many metapopulation models assume that the system has reached extinction-recolonization equilibrium. Variation in demographic rates, numbers of individuals, and other population statistics rarely follow a normal distribution. Effects of threatening processes on populations are rarely linear or log-linear, and threshold effects, in which there are sharp discontinuities in effects, are possible. The effects of multiple threats are often synergistic, rather than additive.

Many of the parameters required to build specific, detailed models of small population dynamics can only be estimated well with long-term and extensive data. It is clearly difficult to obtain large samples from small populations, and conservation action may have to precede long-term field studies in order to ensure that the populations persist long enough to permit extended study. Obtaining a complete enumeration of a small population will remove all sampling error in estimation of current parameter values, but it will not necessarily provide sufficiently precise values for predicting future trends. The entire existing population is still only a sample of the universe of all possible populations that could have resulted from the same processes. Given that highly detailed models employing accurate estimates for a large number of parameters might be needed to project the dynamics of small populations well, should PVA models be used to help guide conservation actions? The alternative to using incomplete models with poorly estimated parameters that may overestimate population viability is to use even more general models that will omit many threatening processes and often more seriously underestimate risks, or to rely on intuitive assessments of complex, probabilistic phenomena, something that people are innately poor at doing (Piattelli-Palmarini 1994, Margolis 1996). When planning conservation actions for species that have already declined to near extinction, we should use the best tools available but also recog-

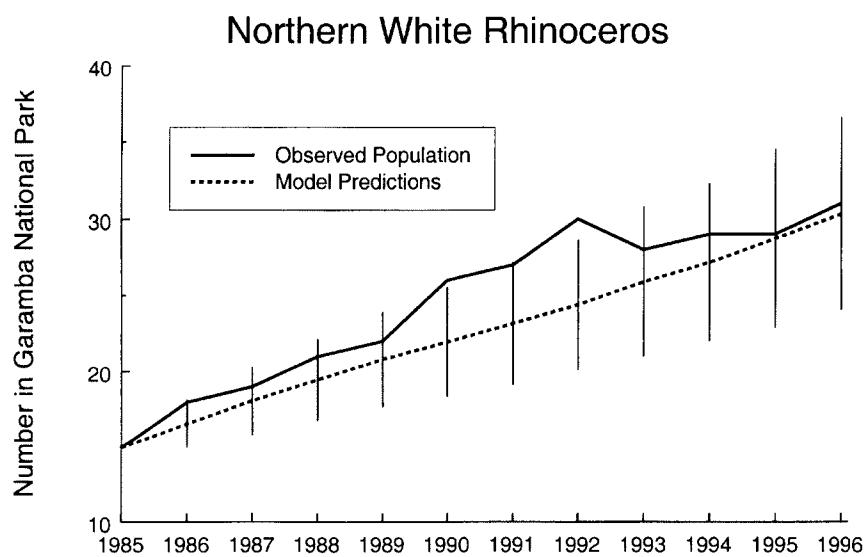


Fig. 5. Observed number (solid line) of northern white rhinoceros in the Garamba National Park, Congo (formerly Zaire), the entire range of the only remaining population of the taxon, and the number projected (dashed line) from the 1985 population. Error bars show the standard deviation across simulations of the projected population size each year. Adapted from Lacy (1996).

nize that our assessments may be crude. Because all PVA models include only a subset of potentially threatening processes, it is possible that many PVAs overestimate viability (Lacy 1993/1994) and that our error will tend to be greatest in the smallest populations – i.e., those which are most critically in need of effective conservation action. Accordingly, margins for error and ongoing monitoring of results should always be part of implementation (see also Akçakaya 2000).

Although the dynamics of small populations can be complex and subjected to many stochastic processes, existing PVA models can provide good representations of dynamics of many such populations. PVA models have predicted well the dynamics of some critically rare species, such as the whooping crane (Brook et al. 1999) and the northern white rhinoceros, *Ceratotherium simum cottoni* (see Fig. 5). Brook et al. (2000) found that PVA models that are sufficiently detailed and for which there are adequate data for estimating parameters can provide unbiased and reasonably accurate predictions for a number of species. The species for which historic population trajectories were modeled in the above studies had been reduced to single populations, rather than existing within metapopulations, and most had relatively simple breeding systems, rather than complex social structures. Also, PVA models may be more reliable when habitat is not limiting population growth than when dynamics near carrying capacity are modeled (Mills et al. 1996, Brook et al. 1997).

How small is small?

The processes and cases described in this paper suggest that, when it comes to assessing whether wildlife populations have declined to dangerously small sizes, small may be bigger than we usually think it is. Isolated populations with fewer than ca 50 breeding adults may suffer from inbreeding depression within a few generations. Larger populations that are fragmented into partially isolated subunits of fewer than ca 50 breeding animals in each may lose variability much faster than would be estimated from the total metapopulation size (Lacy and Lindenmayer 1995). Genetic decay in populations with fewer than ca 500 breeding adults, or even 5000 adults (Lande 1995), may eventually reduce adaptability and potential for recovery (Lacy 1997). Monogamous species and species with complex social systems may have reduced breeding if numbers fall below several hundred adults. Each of these factors tends to be a greater threat in those species, such as many mammals and birds, that have low fecundity and only slow population growth under optimal conditions.

Finally, it may be difficult to know when a population is so small that additional stochastic factors must be included in a PVA to obtain an accurate projection of its dynamics. Therefore, it is often useful to test several models to deter-

mine if added complexity substantially alters PVA predictions, and provides a better fit to observed population trends. A good example of this type of exploration is provided by the ongoing work of Lindenmayer and his colleagues on the fauna of fragmented forests in Australia (Lindenmayer et al. 1999, 2000). PVA models should be no more complex than necessary, but to be useful for conservation they must also be detailed enough to model the real population dynamics accurately (Starfield and Bleloch 1986, Starfield 1997, Akçakaya and Sjögren-Gulve 2000, Lacy and Miller 2001).

Acknowledgements – I thank the editors for their many constructive comments.

References

- Akçakaya, H. R. 2000. Population viability analyses with demographically and spatially structured models. – *Ecol. Bull.* 48: 23–38.
- Akçakaya, H. R. and Ferson, S. 1992. RAMAS/Space: Spatially structured population models for conservation biology. – *Appl. Biomath.*, New York.
- Akçakaya, H. R. and Sjögren-Gulve, P. 2000. Population viability analyses in conservation planning: an overview. – *Ecol. Bull.* 48: 9–21.
- AsRSG. 1996. Report of a meeting of the IUCN SSC Asian Rhino Specialist Group (AsRSG). – Sandakan, Malaysia, 29 Nov. – 1 Dec. 1995.
- Ballou, J. D. 1997. Ancestral inbreeding only minimally affects inbreeding depression in mammalian populations. – *J. Heredity* 88: 169–178.
- Ballou, J. D. et al. 1998. Leontopithecus II. The second population and habitat viability assessment for lion tamarins (*Leontopithecus*). – Conservation Breeding Specialist Group (SSC/IUCN), Apple Valley, Minnesota.
- Barrett, S. C. H. and Kohn, J. R. 1991. Genetic and evolutionary consequences of small population size in plants: implications for conservation. – In: Falk, D. A. and Holsinger, K. E. (eds), *Genetics and conservation of rare plants*. Oxford Univ. Press, pp. 3–30.
- Barton, N. H. and Whitlock, M. C. 1997. The evolution of metapopulations. – In: Hanski, I. A. and Gilpin, M. E. (eds), *Metapopulation biology*. Academic Press, pp. 183–210.
- Belovsky, G. E. 1987. Extinction models and mammalian persistence. – In: Soulé, M. E. (ed.), *Viable populations for conservation*. Cambridge Univ. Press, pp. 35–57.
- Boecklen, W. J. 1986. Optimal reserve design of nature reserves: consequences of genetic drift. – *Biol. Conserv.* 38: 323–338.
- Boyce, M. S. 1992. Population viability analysis. – *Annu. Rev. Ecol. Syst.* 23: 481–506.
- Brewer, B. A. et al. 1990. Inbreeding depression in insular and central populations of *Peromyscus* mice. – *J. Heredity* 81: 257–266.
- Brook, B. W. et al. 1997. Does population viability analysis software predict the behaviour of real populations? A retrospective study on the Lord Howe Island woodhen *Tricholimnas sylvestris* (Slater). – *Biol. Conserv.* 82: 119–128.

- Brook, B. W. et al. 1999. Comparison of the population viability analysis packages GAPPS, INMAT, RAMAS and VORTEX for the whooping crane (*Grus americana*). – *Anim. Conserv.* 2: 23–31.
- Brook, B. W. et al. 2000. Predictive accuracy of population viability analysis in conservation biology. – *Nature* 404: 385–387.
- Brown, J. H. and Kodric-Brown, A. 1977. Turnover rates in insular biogeography: effect of immigration on extinction. – *Ecology* 58: 445–449.
- Caro, T. M. and Laurenson, M. K. 1994. Ecological and genetic factors in conservation: a cautionary tale. – *Science* 263: 485–486.
- Caughley, G. 1994. Directions in conservation biology. – *J. Anim. Ecol.* 63: 215–244.
- Charlesworth, D. and Charlesworth, B. 1987. Inbreeding depression and its evolutionary consequences. – *Annu. Rev. Ecol. Syst.* 18: 237–268.
- Clark, T. W. 1989. Conservation biology of the black-footed ferret. – Wildlife Preservation Trust International, Philadelphia.
- Courchamp, F., Clutton-Brock, T. and Grenfell, B. 1999. Inverse density dependence and the Allee effect. – *Trends Ecol. Evol.* 14: 405–410.
- Dietz, J. M., Baker, A. J. and Ballou, J. D. 2000. Demographic evidence of inbreeding depression in golden lion tamarins. – In: Young, A. and Clarke, G. (eds), *Genetics, demography and population viability*. Cambridge Univ. Press, in press.
- Ellis, S. et al. 1992. Alala, akokohekohe, and palila population and habitat viability assessment reports. – IUCN SSC Captive Breeding Specialist Group, Apple Valley, Minnesota.
- Fahrig, L. and Merriam, G. 1985. Habitat patch connectivity and population survival. – *Ecology* 66: 1762–1768.
- Fahrig, L. and Merriam, G. 1994. Conservation of fragmented populations. – *Conserv. Biol.* 8: 50–59.
- Fisher, R. A. 1958. *The genetical theory of natural selection*. – Dover, New York.
- Foley, P. 1994. Predicting extinction times from environmental stochasticity and carrying capacity. – *Conserv. Biol.* 8: 124–137.
- Forman, L. et al. 1986. Genetic variation within and among lion tamarins. – *Am. J. Phys. Anthropol.* 71: 1–11.
- Frankel, O. H. and Soulé, M. E. 1981. *Conservation and evolution*. – Cambridge Univ. Press.
- Frankham, R. 1995a. Conservation genetics. – *Annu. Rev. Genet.* 29: 305–327.
- Frankham, R. 1995b. Inbreeding and extinction: a threshold effect. – *Conserv. Biol.* 9: 792–799.
- Franklin, J. R. 1980. Evolutionary change in small populations. – In: Soulé, M. E. and Wilcox, B. A. (eds), *Conservation biology: An evolutionary-ecological perspective*. Sinauer, pp. 135–149.
- Gilpin, M. E. and Soulé, M. E. 1986. Minimum viable populations: the processes of species extinction. – In: Soulé, M. E. (ed.), *Conservation biology: the science of scarcity and diversity*. Sinauer, pp. 13–34.
- Goodman, D. 1987. The demography of chance extinction. – In: Soulé, M. E. (ed.), *Viable populations for conservation*. Cambridge Univ. Press, pp. 11–34.
- Gyllenberg, M. and Hanski, I. 1992. Single-species metapopulation dynamics: a structured model. – *Theor. Popul. Biol.* 42: 35–61.
- Hedrick, P. W. 1994. Purgling inbreeding depression and the probability of extinction: full-sib mating. – *Heredity* 73: 363–372.
- Hedrick, P. W. and Miller, P. S. 1992. Conservation genetics: techniques and fundamentals. – *Ecol. Appl.* 2: 30–46.
- Hedrick, P. W. et al. 1996. Directions in conservation biology: comments on Caughley. – *Conserv. Biol.* 10: 1312–1320.
- Jiménez, J. A. et al. 1994. An experimental study of inbreeding depression in a natural habitat. – *Science* 266: 271–273.
- Keane, B. 1990. The effect of relatedness on reproductive success and mate choice in the white-footed mouse, *Peromyscus leucopus*. – *Anim. Behav.* 39: 264–273.
- Keller, L. F. et al. 1994. Selection against inbred song sparrows during a natural population bottleneck. – *Nature* 372: 356–357.
- Kindvall, O. 1996. Habitat heterogeneity and survival of the bush cricket, *Metrioptera bicolor*. – *Ecology* 77: 207–214.
- Lacy, R. C. 1987. Loss of genetic diversity from managed populations: interacting effects of drift, mutation, immigration, selection, and population subdivision. – *Conserv. Biol.* 1: 143–158.
- Lacy, R. C. 1993/1994. What is population (and habitat) viability analysis? – *Primate Conserv.* 14/15: 27–33.
- Lacy, R. 1996. Further population modelling of northern white rhinoceros under various management scenarios. – In: Foose, T. J. (ed.), *Summary, Appendix 3 – Northern White Rhinoceros Conservation Strategy Workshop*. International Rhino Foundation, Cumberland, Ohio, pp. 1–15.
- Lacy, R. C. 1997. Importance of genetic variation to the viability of mammalian populations. – *J. Mammal.* 78: 320–335.
- Lacy, R. C. 2000. Structure of the VORTEX simulation model for population viability analysis. – *Ecol. Bull.* 48: 191–203.
- Lacy, R. C. and Lindenmayer, D. B. 1995. A simulation study of the impacts of population subdivision on the mountain brushtail possum, *Trichosurus caninus* Ogilby (Phalangeridae: Marsupialia), in south-eastern Australia. II. Loss of genetic variation within and between subpopulations. – *Biol. Conserv.* 73: 131–142.
- Lacy, R. C. and Ballou, J. D. 1998. Effectiveness of selection in reducing the genetic load in populations of *Peromyscus polionotus* during generations of inbreeding. – *Evolution* 52: 900–909.
- Lacy, R. C. and Miller, P. S. 2001. Managing the human animal: incorporating human populations and activities into PVA for wildlife conservation. – In: Beissinger, S. and McCullough, D. R. (eds), *Population viability analysis*. Univ. of Chicago Press, in press.
- Lacy, R. C., Alaks, G. and Walsh, A. 1996. Hierarchical analysis of inbreeding depression in *Peromyscus polionotus*. – *Evolution* 50: 2187–2200.
- Lande, R. 1988. Genetics and demography in biological conservation. – *Science* 241: 1455–1460.
- Lande, R. 1995. Mutation and conservation. – *Conserv. Biol.* 9: 782–791.
- Lindenmayer, D. B. and Lacy, R. C. 1995. Metapopulation viability of Leadbeater's possum, *Gymnobelideus leadbeateri*, in fragmented old-growth forests. – *Ecol. Appl.* 5: 164–182.
- Lindenmayer, D. B. et al. 1995. A review of the generic computer programs ALEX, RAMAS/space and VORTEX for modelling the viability of wildlife populations. – *Ecol. Model.* 82: 161–174.

- Lindenmayer, D. B., McCarthy, M. A. and Pope, M. L. 1999. Arboreal marsupial incidence in eucalypt patches in south-eastern Australia: a test of Hanski's incidence function metapopulation model for patch occupancy. – *Oikos* 84: 99–109.
- Lindenmayer, D. B., Lacy, R. C. and Pope, M. L. 2000. Testing a simulation model for population viability analysis. – *Ecol. Appl.* 10: 580–597.
- Lynch, M. and Walsh, B. 1998. Genetics and analysis of quantitative traits. – Sinauer.
- Margolis, H. 1996. Dealing with risk. Why the public and the experts disagree on environmental issues. – Univ. of Chicago Press.
- Menges, E. S. 2000. Applications of population viability analyses in plant conservation. – *Ecol. Bull.* 48: 73–84.
- Miller, P. S. 1994. Is inbreeding depression more severe in a stressful environment? – *Zoo Biol.* 13: 195–208.
- Mills, L. S. and Smouse, P. E. 1994. Demographic consequences of inbreeding in remnant populations. – *Am. Nat.* 114: 412–431.
- Mills, L. S. et al. 1996. Factors leading to different viability predictions for a grizzly bear data set. – *Conserv. Biol.* 10: 863–873.
- Mirande, C., Lacy, R. and Seal, U. 1991. Whooping crane (*Grus americana*) conservation viability assessment workshop report. – IUCN SSC Captive Breeding Specialist Group, Apple Valley, Minnesota.
- Pettersson, B. 1985. Extinction of an isolated population of the middle spotted woodpecker *Dendrocopos medius* (L.) in Sweden and its relation to general theories on extinction. – *Biol. Conserv.* 32: 335–353.
- Piattelli-Palmarini, M. 1994. Inevitable illusions. How mistakes of reason rule our minds. – Wiley.
- Pounds, J. A. and Crump, M. L. 1994. Amphibian declines and climate disturbance: the case of the golden toad and the harlequin frog. – *Conserv. Biol.* 8: 72–85.
- Rabenold, K. N. 1990. *Campylorhynchus* wrens: the ecology of delayed dispersal and cooperation in the Venezuelan savanna. – In: Stacey, P. B. and Koenig, W. D. (eds), Cooperative breeding in birds. Cambridge Univ. Press, pp. 159–196.
- Rabenold, K. N. et al. 1991. Density-dependent dispersal in social wrens: genetic analysis using novel matriline markers. – *Anim. Behav.* 42: 144–146.
- Ralls, K., Ballou, J. D. and Templeton, A. 1988. Estimates of lethal equivalents and the cost of inbreeding in mammals. – *Conserv. Biol.* 2: 185–193.
- Ronce, O., Perret, F. and Olivieri, I. 2000. Evolutionarily stable dispersal rates do not always increase with local extinction rates. – *Am. Nat.* 155: 485–496.
- Ryan, K. K. 2000. Causes and consequences of male mate choice in a monogamous oldfield mouse, *Peromyscus polionotus*. – Ph.D. thesis, Univ. of Chicago.
- Saccheri, I. et al. 1998. Inbreeding and extinction in a butterfly metapopulation. – *Nature* 392: 491–494.
- Schonewald-Cox, C. M. et al. (eds) 1983. Genetics and conservation. A reference for managing wild animal and plant populations. – Benjamin/Cummings, Menlo Park, California.
- Seal, U. S. and Lacy, R. C. 1989. Florida panther population viability analysis. – Report to the U.S. Fish and Wildlife Service. IUCN SSC Captive Breeding Specialist Group, Apple Valley, Minnesota.
- Seal, U. S. et al. 1992. Genetic management strategies and population viability of the Florida panther (*Felis concolor coryi*). – Report to the U.S. Fish and Wildlife Service. IUCN SSC Captive Breeding Specialist Group, Apple Valley, Minnesota.
- Shaffer, M. L. 1981. Minimum population sizes for species conservation. – *BioScience* 31: 131–134.
- Shaffer, M. 1987. Minimum viable populations: coping with uncertainty. – In: Soulé, M. E. (ed.), Viable populations for conservation. Cambridge Univ. Press, pp. 69–86.
- Sjögren-Gulve, P. 1994. Distribution and extinction patterns within a northern metapopulation of the pool frog, *Rana lessonae*. – *Ecology* 75: 1357–1367.
- Sjögren-Gulve, P. and Ray, C. 1996. Using logistic regression to model metapopulation dynamics: Large-scale forestry extirpates the pool frog. – In: McCullough, D. R. (ed.), Metapopulations and wildlife conservation. Island Press, Washington, D.C., pp. 111–137.
- Soulé, M. E. 1980. Thresholds for survival: maintaining fitness and evolutionary potential. – In: Soulé, M. E. and Wilcox, B. A. (eds), Conservation biology. An evolutionary-ecological perspective. Sinauer, pp. 151–169.
- Soulé, M. E. 1987. Viable populations for conservation. – Cambridge Univ. Press.
- Starfield, A. M. 1997. A pragmatic approach to modeling. – *J. Wildl. Manage.* 61: 261–270.
- Starfield, A. M. and Bleloch, A. L. 1986. Building models for conservation and wildlife management. – MacMillan, New York.
- Stephens, P. A. and Sutherland, W. J. 1999. Consequences of the Allee effect for behaviour, ecology and conservation. – *Trends Ecol. Evol.* 14: 401–405.
- Varvio, S. L., Chakraborty, R. and Nei, M. 1986. Genetic variation in subdivided populations and conservation genetics. – *Heredity* 57: 189–198.
- Vucetich, J. A. and Creel, S. 1999. Ecological interactions, social organization, and extinction risk in African wild dogs. – *Conserv. Biol.* 13: 1172–1182.
- Vucetich, J. A., Peterson, R. O. and Waite, T. A. 1997. Effects of social structure and prey dynamics on extinction risk in gray wolves. – *Conserv. Biol.* 11: 957–965.
- Westermeier, R. L. et al. 1998. Tracking the long-term decline and recovery of an isolated population. – *Science* 282: 1695–1698.